CONSERVATOIRE NATIONAL DES ARTS ET METIERS

PARIS

MEMOIRE

Présenté en vue d'obtenir

LE DIPLOME D'INGENIEUR C.N.A.M.

en

ELECTRONIQUE

par

Philippe MARTIN

ETUDE ET REALISATION D'UN TERMINAL DE VISUALISATION ALPHANUMERIQUE

- FONCTIONS ET LOGIQUE DE COMMANDE -

Soutenu le 06 MAI 1981

JURY

PRESIDENT : Monsieur le Professeur ANGEL

MEMBRES : MM. VABRE, JAFFRAIN, BALLA

Qu'il me soit permis d'exprimer,

Ma gratitude,

à Messieurs les Professeurs du Conservatoire National des Arts et Métiers pour les enseignements qu'ils m'ont dispensés

Ma reconnaissance,

à Messieurs MESCAM et MARGERIN, mes patrons à la CII-HONEYWELL-BULL, qui m'ont permis d'effectuer cette étude en parfaite harmonie avec les impératifs du service.

Mes remerciements,

à Mesdames DELPY, et GIORDMAINA qui ont transcrit ce mémoire avec beaucoup de soins et de gentillesse.

INTRODUCTION

1. SYSTEME INFORMATIQUE ET TERMINAL DE VISUALISATION

1.1. Terminal de visualisation pour un système informatique

1.1.2. Contexte matériel - architecture système
- poste de travail

1.1.3. Contexte logiciel

1.2. Logique de commande pour un terminal de visualisation

2. INTRODUCTION A L'ETUDE

2.1. Etat actuel des terminaux de visualisation

2.2. Objectifs du projet

2.2.1. Facteurs humains

2.2.2. Objectifs fonctionnels

2.2.3. Objectifs économiques

FONCTIONS ORIGINALES DU TERMINAL

1. FONCTIONS ERGONOMIQUES

1.1. Surface de visualisation

1.2. Surface des caractères affichables

1.3. Attributs visuels

1.4. Niveaux d'intensité lumineuse et sonore

2. FONCTIONS MATERIELLES

2.1. Interface de la ligne de communication

2.2. Interfaces périphériques

2.3. La visualisation

3. FONCTIONS LOGICIELLES

3.1. Synthèse des fonctions actuelles

3.1.1. Procédures de transmission

3.1.2. Protocoles de commande

3.2. Fonctions nouvelles

LOGIQUE DE COMMANDE : SOLUTIONS POSSIBLES ET CHOIX

1. PRINCIPE D'ARCHITECTURE DE LA LOGIQUE DE COMMANDE

1.1. L'unité centrale : le microprocesseur

1.2. La mémoire

1.3. Les circuits périphériques

1.4. Les bus d'interconnexions

2. LOGIQUE DES INTERFACES OPTIONNELS

2.1. Principes de fonctionnement

2.2. Application à l'interface " lecteur de badges "

2.3. Comparaison des solutions et choix.

3. CHOIX DE LA LOGIQUE DE VISUALISATION

3.1. Principes de réalisation du rafraîchissement de l'image

3.1.1. Rappel

3.1.2. Mémoire d'écran spécialisée

3.1.3. Mémoire d'écran banalisée

3.1.4. Autre principe

3.2. Comparaison des deux techniques

3.3. Deux projets de réalisation de la logique de visualisation

3.3.1. Mémoire d'écran spécialisé

3.3.2. Mémoire d'écran banalisée

3.4. Comparaison et choix

3.4.1. Performances

3.4.2. Coût en matériel

3.4.3. Fonctions remplies

3.4.4. Conclusion

4. CHOIX DES MEMOIRES

4.1. La mémoire de commande

4.1.1. Mémoire morte

4.1.2. Mémoire vive

4.2. La mémoire du générateur des caractères

4.3. Mémoire de travail et mémoire d'écran

4.4. La mémoire de sauvegarde

5. CHOIX DU CONTROLEUR D'ACCES DIRECT MEMOIRE (DMA)

5.1. Rappel

5.2. Principe de fonctionnement du DMA

5.3. Deux types de DMA

5.4. Performances et choix

LOGIQUE DE COMMANDE : FONCTIONNEMENT (HARDWARE)

1. ADRESSAGE DES MEMOIRES ET PERIPHERIQUES

2. TRANSFERTS DE L'INFORMATION

2.1. Différents types de transferts

2.1.1. Transferts gérés par le microprocesseur

2.1.2. Transferts gérés par le DMA

2.2. Cycles de lecture et d'écriture du microprocesseur

2.3. Cycles de lecture et d'écriture du DMA

3. LA VISUALISATION

3.1. Lecture et écriture en mémoire d'écran

3.2. Rafraîchissement de l'image

3.3. Horloges de visualisation

3.4. Le curseur et les attributs visuels

4. LES FONCTIONS NOUVELLES

4.1. " Multi-page " et " Partitionnement d'écran "

4.2. Réglages des niveaux de luminosité et d'alarme sonore

4.3. Sauvegarde d'information à la mise hors tension de l'appareil.

CONCLUSION

INDEX DES ABREVIATIONS

CII-HB	:	Compagnie Internationale pour 1' Informatique Honeywell Bull
CCITT	:	Comité Consultatif International pour le Télégraphe et le Téléphone.
ISO	:	International Standard Organisation
ECMA	:	European Computer Manufactured Association
ASCII	:	American Standard Code Information Interchange
HDLC	:	High Level Data Link Control
CRT	:	Cathod Ray Tube
MODEM	:	Modulateur - Demodulateur
BUS	:	Réseau de liaisons conductrices
TTL	:	Transistor transistor Logic
MOS	:	Metallic Oxyd semi-conductor
MSI	:	Medium Scale Integration
LSI	:	Large Scale Integration
EPROM	:	Electrically Programmable Read Only Memory
RAM	:	Random Access Memory
DMA	:	Direct Memory Access
USART	:	Universal Synchronous/Asynchronous Receiver Transmitter
PPI	:	Programmable Peripheral Interface
PIT	:	Programmable Internal Timer
FIFO	:	First In First Out
UPI	:	Universal Peripheral Interface
kø	:	Kilo - octet : 1024 mots de 8 bits (8192 Bits)
bps	:	Bits par seconde

INTRODUCTION

1. SYSTEME INFORMATIQUE ET TERMINAL DE VISUALISATION

. Définition de l 'appareil

L'appareil, dont un élément fait l'objet de ce mémoire, est un terminal de visualisation alphanumérique.

Le terminal est un périphérique d'ordinateur central qui permet la saisie et le transfert d'informations à distance.

Ce terminal est dit de visualisation parce qu'il affiche des informations, reçues ou à émettre, sur l'écran d'un tube à rayons cathodiques.

La visualisation est alphanumérique parce que des lettres, des chiffres ou des symboles bien définis peuvent être affichés.

1.1.- TERMINAL DE VISUALISATION POUR UN SYSTEME INFORMATIQUE

1.1.1. Contexte matériel

1.1.1.1. Architecture de système

Le produit est co nçu pour être intégré dans u n système de type télé-informatique ; il accède aux informations contenues dans une banque de données, gérée par un ordinateur central, et il fournit à cette banque de nouvelles informations introduites par un opérateur ; c'est la notion de saisie de données.

D'une application à une autre, l'architecture d'un système de communication peut prendre des formes très différentes. Le nombre de terminaux et leur distance avec l'ordinateur central caractérisent la structure du système.

Le système le plus simple est local et composé d'un terminal relié directement à un ordinateur.

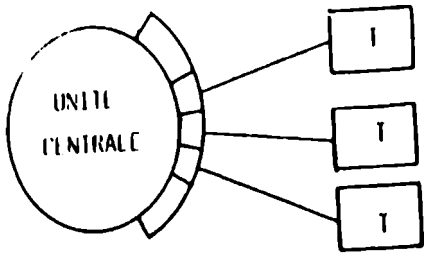
La figure 1. représente l'architecture d'un système complexe qui fait apparaitre trois ensembles :

- le système central
- le système terminal
- le support de transmission longue distance (facultatif)

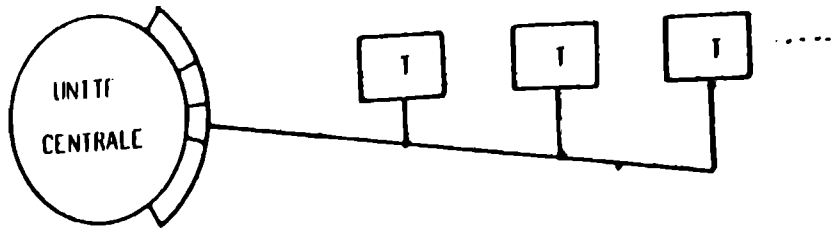
Le système terminal, dans lequel s'intègre l'appareil étudié, se compose d'un contrôleur et de terminaux reliés en structure d'étoile ou de type multipoint.

Le contrôleur multiplexe les informations en provenance de l'ordinateur et éventuellement adapte les langages de communication.

Le terminal constitue l'organe de dialogue avec un opérateur. Il peut à son niveau commander des sous-ensembles tels qu'une imprimante, et constitue alors un poste de travail.



STRUCTURE ETOILEE



STRUCTURE MULTIPOINT

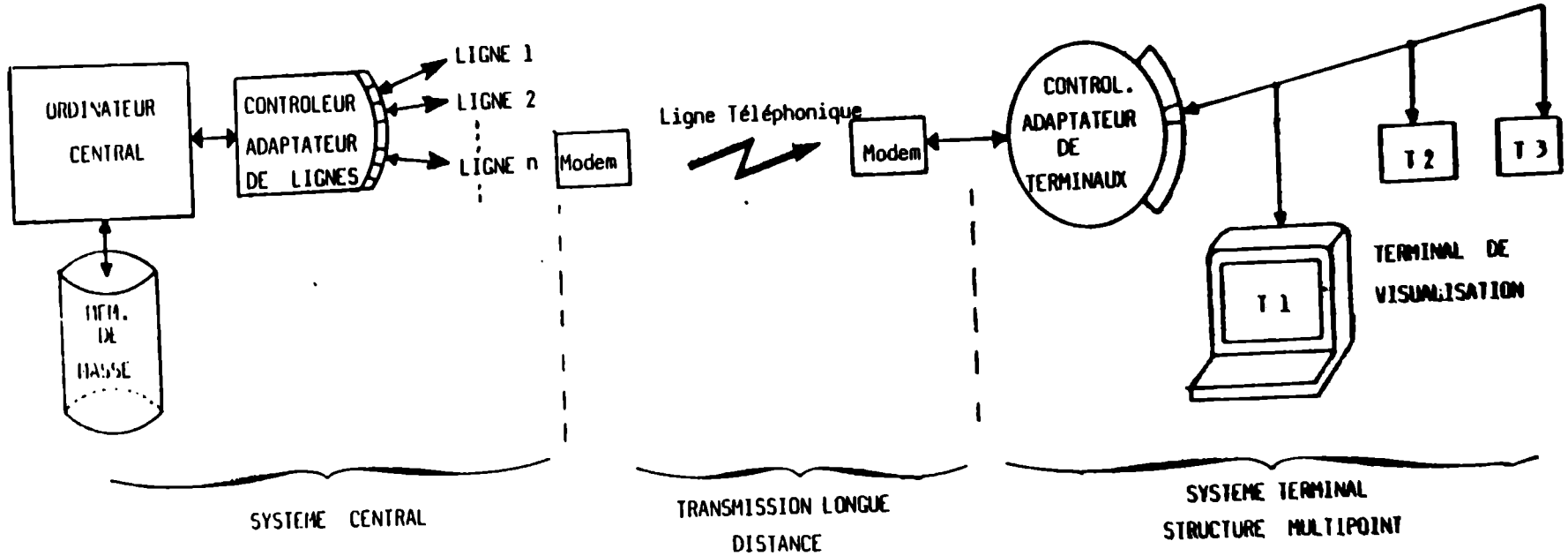
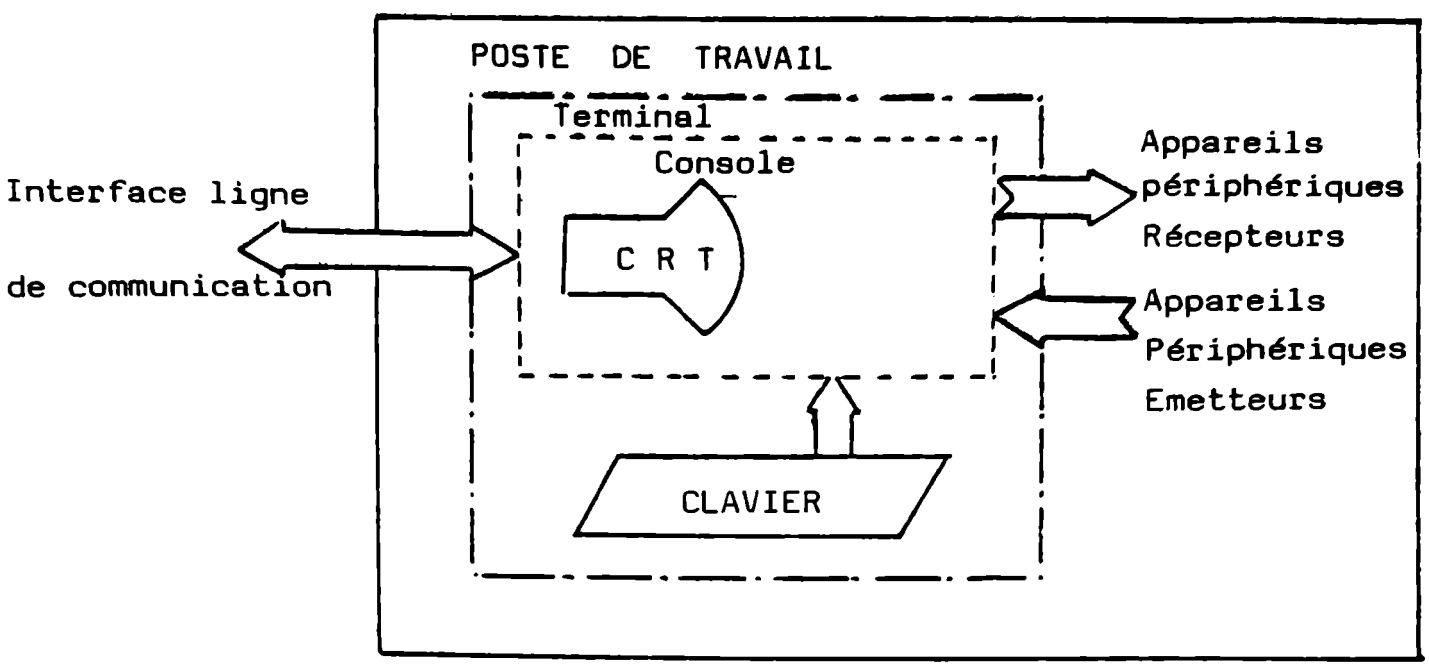


Figure 1 : EXEMPLE D' ARCHITECTURE

1.1.1.2. Poste de travail



Le terminal destiné à des applications de 'saisies' gère des réceptions et des émissions de données ; il est constitué d'une console de visualisation et d'un clavier.

Localement, la console de visualisation est un système utilisant un microprocesseur qui gère une mémoire et des sous-ensembles tels que l'écran CRT et le clavier qui constituent le terminal proprement dit, et d'autres appareils externes dits appareils périphériques (imprimantes, lecteurs, ...). L'ensemble constitue un poste de travail.

Les informations sont visualisées sur l'écran et celles qui doivent être conservées peuvent être imprimées ; les saisies de données sont effectuées à partir du clavier ou des lecteurs, et l'information est généralement visualisée avant (ou en même temps) qu'elle soit émise sur la ligne de communication.

L'interface physique vers la ligne de communication est caractérisé par le type de transmission simultanée (full-duplex) ou à l'alternat (half duplex), en mode asynchrone ou synchrone.

1.1.2. Contexte logiciel

Le traitement de l'information dans un système définit le logiciel. Les modes de traitement diffèrent suivant les applications et les critères de choix sont de deux types :

- critères de téléinformatique
- critères de transmission

1.1.2.1. Les critères de téléinformatique

Ce sont principalement la gestion en ' temps partagé ' et la gestion en ' temps réel ' .

- La gestion en temps partagé est un mode conversationnel dans lequel l'ordinateur interroge et autorise à des moments précis le terminal pour qu'il envoie ses informations. C'est une procédure par scrutation.

- La gestion en temps réel est un mode dans lequel le terminal demande la main et monopolise l'ordinateur toute la durée d'une transaction. C'est une procédure par interruption.

1.1.2.2. Les critères de transmission

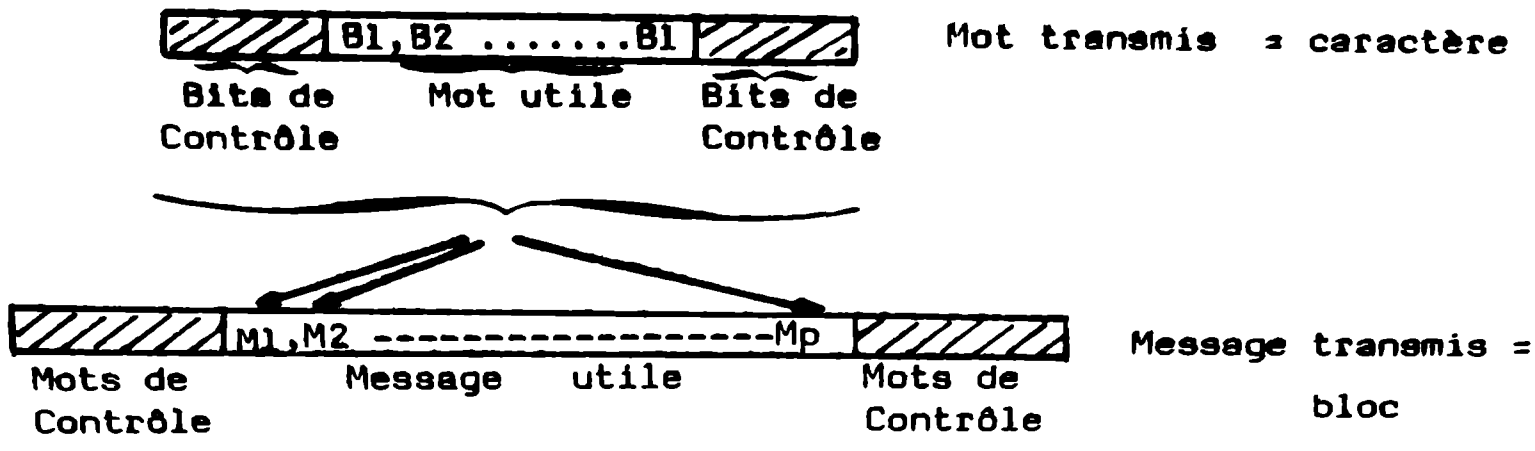
Quel que soit le mode de traitement utilisé le logiciel doit réaliser deux fonctions :

- régler le dialogue ordinateur - terminal en évitant les conflits par une procédure de dialogue

- parer aux erreurs amenées par la transmission

Au niveau du terminal ceci apparaît dans la structure de l'information transmise.

. **Structure de l'information**



Le terminal peut utiliser des procédures de dialogue différentes suivant le microprogramme avec lequel il est équipé. L'information est structurée en mots (plusieurs bits jointifs) et en message (plusieurs mots jointifs) qui comprennent des bits utiles (données) et des bits de contrôle de transmission.

. En mode ' caractère ' le transfert s'effectue caractère par caractère sans notion de message.

En mode ' bloc ' le transfert s'effectue par messages. Le message le plus simple comprend un mot de début, un mot de fin et éventuellement un mot de contrôle d'erreurs.

Les messages sont plus complexes lorsque le système utilise une procédure de dialogue ; certains de ces messages contiennent alors uniquement de l'information de contrôle (messages d'interrogation, messages d'état, acquittements).

Contrôle des erreurs

Les transmissions à distance utilisant les lignes téléphoniques subissent des perturbations qui rendent nécessaire un contrôle d'erreur. La détection s'effectue à 2 niveaux :

- au niveau du mot par une parité transversale (paire ou impaire)
- au niveau du message par une parité longitudinale (paire ou impaire) ou par un mot résultant d'un calcul mathématique (codes cycliques).

La correction des erreurs s'effectue généralement par la réémission du message.

En résumé, le contexte logiciel se traduit au niveau de l'appareil par la réalisation des fonctions suivantes :

. Modes de transmission :

- simultanée ou à l'alternat
- asynchrone ou synchrone
- par caractères ou par blocs de caractères
- avec ou sans procédure de dialogue
- avec contrôle d'erreur par parité ou par codes

Fréquences de transmission

- de 50 à 1200 bps pour les liaison téléphoniques du public (bande passante de 300 Hz à 3 KHz)

- jusqu'à 9600 bps pour les liaisons téléphoniques louées

- jusqu'à 100.000 bps pour les liaisons directes

1.2. LA LOGIQUE DE COMMANDE POUR UN TERMINAL DE VISUALISATION.

L'étude qui fait l'objet de ce mémoire se limite à la logique et au matériel nécessaires pour commander le fonctionnement du terminal, appelés par la suite ' logique de commande '.

Cette logique est principalement composée de circuits intégrés implantés sur un ou plusieurs circuits imprimés.

Le traitement des signaux est numérique.

La figure 2 montre la structure du terminal .

L'appareil est constitué de 2 ensembles :

le clavier
la console

. Le clavier contient les touches et la logique nécessaire à la détection des appuis et à l'envoi des signaux vers l'interface clavier.

. La console renferme trois éléments :

un bloc d'alimentation
un ensemble moniteur de vidéo
une carte de logique de commande de base et
des cartes de logique de commande des interfaces périphériques optionnels.

* L'ensemble d'alimentation fournit les tensions utilisées par les circuits logiques et certains appareils périphériques. La puissance disponible est de 35 VA.

* L'ensemble moniteur, directement alimenté par le secteur, se compose d'une carte à circuit imprimé (le moniteur) et d'un tube à rayons cathodiques.

La carte moniteur réalise, à partir des signaux numériques de synchronisation et de vidéo fournis par la logique de commande, l'amplification et la génération des signaux nécessaires au fonctionnement du tube :

- signal vidéo de cathode
- rampes de courant de déviation horizontale et verticale du faisceau
- signaux de correction de linéarité et de focalisation
- haute tension

* Les cartes de logique de commande traitent toutes les informations reçues ou à émettre sur les interfaces tels que ceux de la ligne de communication, de l'écran, du clavier et éventuellement d'une imprimante, d'un lecteur, d'un photostyle, etc .. Ceci est réalisé à partir des ensembles suivants :

- une unité centrale qui sert de processeur à l'intérieur du terminal
- une mémoire qui contient le programme nécessaire au fonctionnement
- une mémoire qui contient les informations à visualiser sur l'écran
- la logique nécessaire à la visualisation sur l'écran
- les circuits d'interfaces logiques .

Cet ensemble logique constitue le sujet de l'étude développée par la suite.

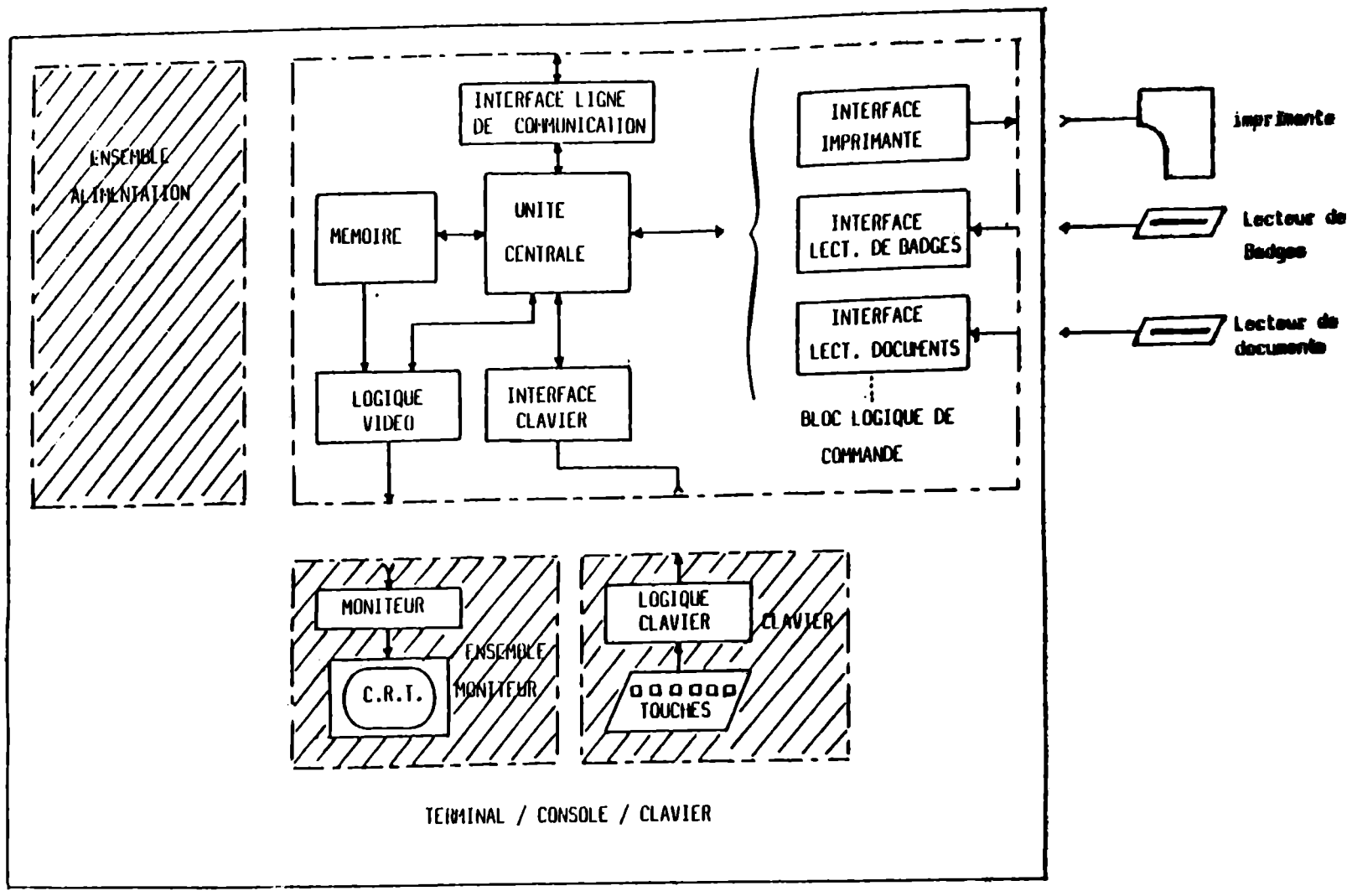


Figure 2 : ARCHITECTURE DU TERMINAL

2. INTRODUCTION A L'ETUDE

2.1. Etat actuel des terminaux de visualisation

Les fonctions remplies par ce type de terminal sont en constante évolution. La tendance générale, dans les systèmes de gestion, est de décentraliser l'intelligence de l'ordinateur central vers les terminaux. Des fonctions nouvelles sont aussi ajoutées à cause du développement que prennent certaines applications comme le traitement de texte. Toutefois les grandes lignes qui caractérisent l'architecture et la gestion du terminal sont établies depuis plusieurs générations d'appareils et ne sont pas remises en cause.

Avant l'avènement des microprocesseurs, le matériel était réalisé, à partir de circuits intégrés, en logique cablée (sans programmation). Aujourd'hui il est organisé à l'aide d'un microprocesseur et d'autres circuits en partie programmables, qui permettent d'utiliser un même matériel pour des applications différentes.

La fonction principale, la visualisation, est réalisée sur l'écran d'un tube à rayons cathodiques. Bien que sur le plan technologique, d'autres types d'écrans (écrans à plasma, écrans à état solide) viennent concurrencer l'écran type CRT, sur le plan industriel et commercial, ce dernier a de loin la préférence (industrialisation très poussée grâce à la télévision).

2.2. Objectifs du projet

Seuls les objectifs ayant une incidence dans le choix de la logique de commande sont décrits.

2.2.1. Les facteurs humains et ergonomiques

- Taille de l'écran : 15 pouces ;
- Amélioratìons des caractèristiques visuelles :
  - Photométrie, forme et dimensions des caractères, nouvelles fonctions de visualisation.
- Contrôle par l'opérateur des niveaux d'intensité sonore et lumineuse.

2.2.2. Les objectifs fonctionnels :

Le terminal, appelé " Visuel Unifié " (VU), doit assurer les fonctions (émulation)* des terminaux de visualisation déjà utilisés à la CII-HB, mais aussi réaliser de nouvelles fonctions.

. Emulation de quatre terminaux existants :

- Un terminal asynchrone à mode caractère (VU1)
- Un terminal asynchrone à mode caractère et bloc (VU2)
- Un terminal synchrone à mode bloc, à procédure type HDLC et à protocole de commande chargeable en mémoire à partir de la ligne de communication (VU3).
- Un terminal synchrone, à mode bloc et à procédure type VIP (VU4).

L'émulation des terminaux VU1, VU2 et VU4 est réalisée avec un matériel unique ; seul le microprogramme de commande implanté dans la mémoire change. L'émulation du terminal VU3 nécessite un matériel légèrement différent.

. Pour être compétitif dans les années 1981 à 1985, le terminal doit également réaliser de nouvelles fonctions logiques.

2.2.3. Les objectifs économiques :

L'appareil est destiné à être fabriqué en très grandes quantités(par dizaines de milliers) et il touche le marché des terminaux de visualisation milieu de gamme.Tout en assurant davantage de fonctions et une meilleure qualité visuelle que les appareils actuels, il doit coûter moins cher (objectif de 4000 F pour 1982).

Les objectifs ainsi définis, mènent à l'étude d'une nouvelle logique de commande, en particulier en ce qui concerne la visualisation, mais aussi la réalisation des fonctions nouvelles qui ont un impact sur le matériel.

* Réaliser ' l'émulation ' fonctionnelle d'un appareil consiste à fournir un second appareil qui assume au moins toutes les fonctions du premier.

FONCTIONS ORIGINALES DU TERMINAL

Le produit étudié est original dans tous ses éléments car, aussi bien l'alimentation, que le type de clavier ou le type d'écran CRT font l'objet d'études nouvelles. Dans ce mémoire, seuls les problèmes qui concernent la logique de commande sont abordés. Ce chapitre développe les fonctions originales que doit réaliser cette logique, celles qui amènent de nouvelles études.

Le problème du choix du type de microprocesseur, par exemple, ne sera pas abordé ; ce circuit, le 8085-A de chez Intel, étant connu, déjà utilisé dans des études précédentes et non remis en cause par de nouveaux concurrents.

1. FONCTIONS ERGONOMIQUES

Ce sont entre autres, les fonctions qui vont dans le sens d'une meilleure adaptation de l'appareil à l'utilisateur, celles qui améliorent la qualité d'image et la facilité des interventions de l'opérateur. Elles apparaissent au niveau de la mécanique, de la logique de commande et du microprogramme seul l'aspect logique de commande est décrit.

1.1. Surface de visualisation

L'écran est du type 15 pouces (38 cm de la diagonale de la surface de phosphore) afin d'offrir à l'opérateur un angle de vision de l'écran adapté à certains postes de travail.

Les générations précédentes de consoles étant de type 12 pouces (30,5 cm), la logique de commande doit gèrer un nombre de points vidéo plus élevés et créer des signaux de visualisation adaptés au tube.

1.2. Surface des caractères affichables

Le terminal, à usage purement informatique, visualise une suite de caractères alphanumériques et symboliques, dont le nombre peut atteindre 2000 (25 lignes de 80 caractères). La surface de l'écran peut donc être virtuellement découpée en 2000 éléments de surface (fenêtres), chacune comprenant une zone visualisable : la matrice de caractères (fig.3).

La qualité de visualisation est liée au nombre de points vidéo que peut contenir cette matrice. Il est de 8 points horizontalement et de 11 points verticalement, contre 5 à 7 par 7 à 9 pour les générations antérieures de terminaux.

Les interlignes et inter-colonnes permettent de séparer les caractères.

Certains caractères spéciaux (caractères graphiques) permettent de visualiser des lignes horizontales ou verticales continues, dans ce cas, des points peuvent être visualisés à l'extérieur de la matrice.

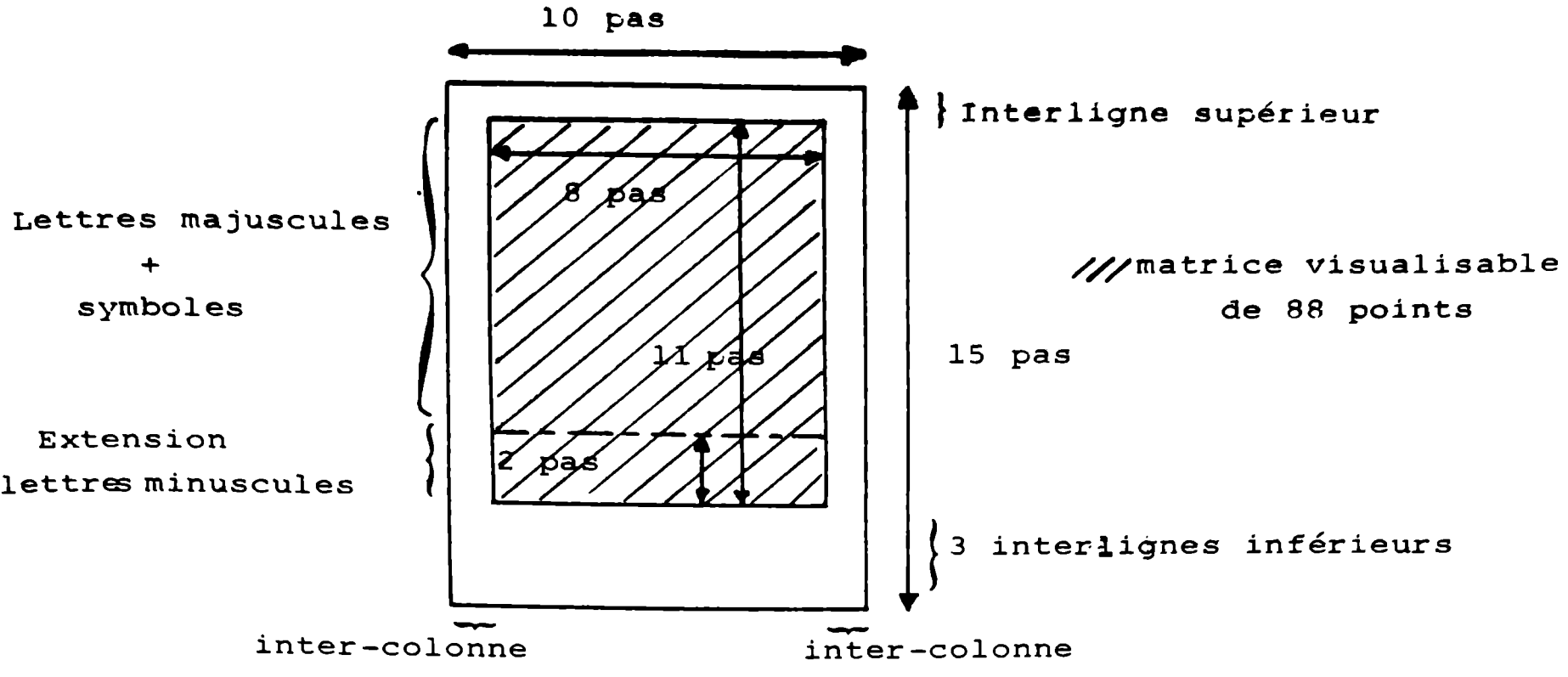


FIGURE 3 : FENETRE DE CARACTERE

1.3. Attributs visuels :

Pour faciliter le dialogue entre l'ordinateur et un opérateur (mise en évidence de titres, de messages d'alarme, de zones à saisir)

la logique de commande réalise des fonctions purement visuelles, telles que :

- Le clignotement ;
- Le soulignement ;
- La luminosité réduite ;
- L'inversion vidéo (caractères noirs sur le fond lumineux);
- La colonne de séparation.

Les commandes correspondantes sont appelées " attributs de visualisation ", auxquelles il faut ajouter l'attribut " champ secret " qui masque l'affichage du caractère affecté.

Une amélioration notoire par rapport aux terminaux actuels consiste à ne pas limiter le nombre des attributs visuels (2000 attributs maximum associés aux 2000 caractères).

1.4. Niveaux d'intensité lumineuse et sonore, sélections techniques

- Pour améliorer la souplesse d'utilisation, le terminal rend accessible à l'opérateur une fourchette de règlage des niveaux d'intensité lumineuse et sonore.
  Ces règlages sont réalisés à partir du clavier et doivent donc être traités par la logique de commande. Un premier règlage de luminosité définit le niveau d'intensité normale et un second le contraste entre niveau d'intensité normale et d'intensité réduite.

  Le signal sonore peut être règlé en intensité et en fréquence.

  Ces réglages étant effectués en chargeant des données à partir du clavier, il est nécessaire que ces données soient mémorisées par un système de sauvegarde à la mise hors tension de l'appareil.

- Les sélections techniques sont des interrupteurs logiques qui permettent de paramètrer le mode de fonctionnement du terminal. Elles sont positionnées par l'opérateur à partir du clavier et doivent également être sauvegardées en cas de mise hors tension.

  A la mise sous tension toutes ces informations sont restituées.

2. FONCTIONS MATERIELLES (HARDWARE)

Les fonctions dites " matérielles " sont réalisées à partir de circuits spécialisés. Ce sont des fonctions normalisées (transmission sur une ligne) ou d'aspect répétitif (rafraîchissement de l'image d'écran).

2.1. L'interface de la ligne de communication

Il réalise la mise en forme des octets de données à émettre et le décodage des données reçues, en respectant le type de transmission sélectionné dont les paramètres sont :

- Liaison à détection de niveau de tension, de type V 24, pour les courtes distances (< 5 m) ou liaison à détection différentielle de tension, de type V 11, pour les moyennes distances (de l'ordre du km).
- Transmission simultanée ou à l'alternat
- Transmission synchrone ou asynchrone
- Vitesses de transmission de 50 à 19200 bps pour la liaison V 24 et jusqu'à 100 000 bps pour la liaison V 11.

2.2. Les interfaces périphériques

- L'interface du clavier prélève et mémorise les données (codes des touches) émises par le clavier .
- Les interfaces des appareils périphériques sont réalisés sur des cartes logiques séparées et optionnelles.
  Une carte est spécifique d'un type d'appareil (exemple : l'interface imprimante) mais permet de connecter différents modèles du même type (plusieurs modèles d'imprimantes par exemple).

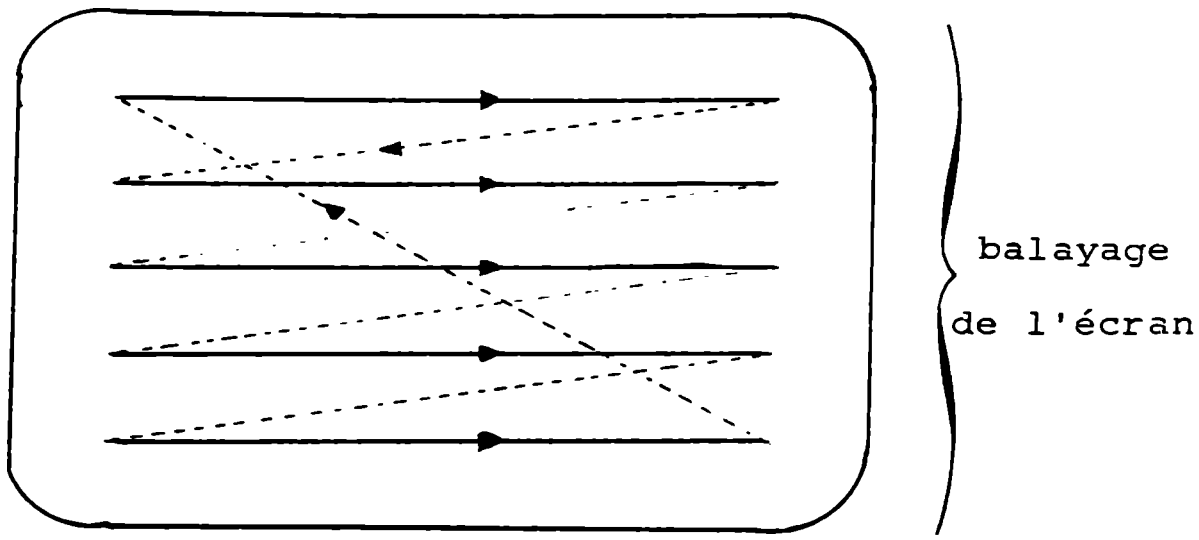
2.3. La visualisation

2.3.1. La mémoire

La fonction mémoire d'écran ne se réduit pas à un problème de mémoire d'image, car il faut pouvoir non seulement transformer un message reçu de la ligne du clavier ou d'un appareil périphérique en image, mais également extraire de l'image un message envoyé vers l'ordinateur central.

En pratique la mémoire contient les codes des caractères et des attributs visuels du texte visualisé.

2.3.2. Le rafraîchissement de l'image



Le contenu de la mémoire est exploré cycliquement de façon à répéter à une fréquence suffisante l'inscription de l'image sur l'écran.

Le balayage du faisceau électronique sur l'écran est horizontal de type télévision non entrelacé. Le rafraîchissement de l'image (balayage de l'écran entier) s'effectue à une fréquence de 50 HZ.

Cette fréquence doit en effet être de 50 HZ au moins pour assurer à l'oeil une image sans scintillement avec un phosphore d'écran de rémanence de 0, 002 à 0,06 ms.

Cette fréquence est fixée à 50 HZ (fréquence du secteur) pour éviter les effets de moirage provoqués par l'ondulation résiduelle sur l'alimentation et les champs magnétiques internes au terminal.

2.3.3. L'affichage

Le terminal doit pouvoir afficher 2000 caractères en utilisant 256 graphismes différents. Les terminaux existant déjà n'utilisent que 96 graphismes (44 symboles et chiffres, etles alphabets, majuscule et minuscule).

Le générateur de caractères utilise les codes de caractères issus de l'exploration de la mémoire et les traduit en commandes d'allumage du spot du CRT. C'est une mémoire de transcodage dont la capacité et le contenu peuvent être modifiés.

L'extension du générateur de caractères permet d'y introduire les lettres particulières des alphabets nationaux, les symboles de ' télétexte ' (utilisés en traitement de texte) de ' datascope ' (visualisation sous forme mnémonique d'un code de commande) et les symboles graphiques permettant d'afficher des figures.

3. FONCTIONS LOGICIELLES (FIRMWARE)

Les fonctions dites ' logicielles ' sont réalisées à partir d'un microprogramme activé par des évènements extérieurs. Elles permettent de traiter les informations reçues par le terminal.

Deux niveaux de fonctions sont distincts :

- Les fonctions qui traitent le dialogue de transmission avec l'unité centrale : c'est la procédure de dialogue ;

- Les fonctions qui traitent les commandes propres au terminal : c'est le protocole de commande.

La plupart de ces fonctions sont réalisées par le microprogramme, le microprocesseur et une mémoire de travail; d'autres nécessitent le développement d'une logique de commande spécifique.

3.1. Synthèse des fonctions actuelles

C'est l'ensemble des fonctions réalisées par les quatre terminaux à émuler.

3.1.1. Procédures de transmission

Le terminal équipé d'un programme type VU 1 ou VU 2 n'utilise pas de procédure de dialogue (pas de liaison ' multipoint ').

Le terminal équipé d'un programme type VU 3 ou VU 4, utilise soit la procédure HDLC, soit la procédure VIP. Toutes deux, de même principe, sont utilisées dans des liaisons ' multipoint ' entre terminaux.

L'unité centrale garde l'initiative en interrogeant successivement chaque terminal, identifié par une adresse, pour savoir si il est prêt à émettre ou à recevoir des données.

Le dialogue est ainsi constitué à partir de trois types de messages :

- message de sélection d'un terminal (selecting)

- message d'interrogation d'un terminal (polling)

- message de données, d'état, ou d'accusé de réception.

Des circuits spécialisés permettent de gérer en partie ces procédures.

3.1.2. Protocole de commande

Le format des mots utilisés en télétransmission est fixé à 7 bits utiles par la norme ISO 646 (ASCII), ce qui limite à 128 le nombre de combinaisons. 96 d'entre elles sont réservées pour les caractères affichables et 32 pour les commandes.

Toutefois l'augmentation du degré d'intelligence des terminaux oblige à étendre le nombre de commandes en utilisant plusieurs mots pour coder une seule commande. Le codage, précisé par les normes ECMA, définit des commandes ' bi-codes ' et ' multicodes '.

	Commandes 'Monocode' (ISO)	Commandes 'Bicodes' (ECMA)	Commandes 'Multi-codes (ECMA)
Nbre de commandes possibles	32-1 introducteurs $(00)_{16}$ à $(1F)_{16}$	128-2 introducteurs $(00)_{16}$ à $(FF)_{16}$	illimité P1;P2;...Pn + $\{(00)_{16}$ $(FF)_{16}$
Introducteur	ESCAPE $(1B)_{16}$	DCS $(50)_{16}$] } CSI $(5B)_{16}$]	

Les commandes traitées par le terminal sont regroupées par familles; ce sont :

- Les commandes de formatage qui permettent de définir sur un écran un format à exploiter par l'opérateur (un formulaire par exemple).
- Les commandes d'édition nécessaires à la mise en page du texte affiché
- Les commandes d'effacement.
- Les commandes de contrôle (contrôle du texte affiché, du clavier, de l'état ou de la configuration du terminal, du transfert des données etc...).
- Et les commandes d'impression ou de lecture d'un message émis par un appareil périphérique.

La liste exhaustive du protocole de commande de chaque terminal à émuler est donnée en annexe (1).

3.2. Les fonctions nouvelles

La plupart des fonctions nouvelles réalisées par la logique de commande sont traitées par le microprogramme sans conséquence sur le matériel.

Deux nouvelles fonctions à réaliser vont toutefois influencer le choix du matériel de la logique, ce sont :

- L'extension quantitative et qualitative des attributs de visualisation (cf. paragraphe 2.1.3.).
- Les fonctionnements " multipage " et " partitionnement ".

. Fonctionnement ' multi-page ' et ' partitionnement ' :

Dans le but d'alléger la tâche de l'ordinateur central, il sera offert à l'opérateur de travailler sur un espace plus grand que l'espace mémoire visualisé sur l'écran.

Définitions :

- Une " page " est l'espace mémoire correspondant à un écran visualisé.
- Un terminal de visualisation est dit " multi-page " si il peut gérer plusieurs pages de données , une seule étant visualisée et accessible à un instant donné.
- Partitionnement de la mémoire écran :

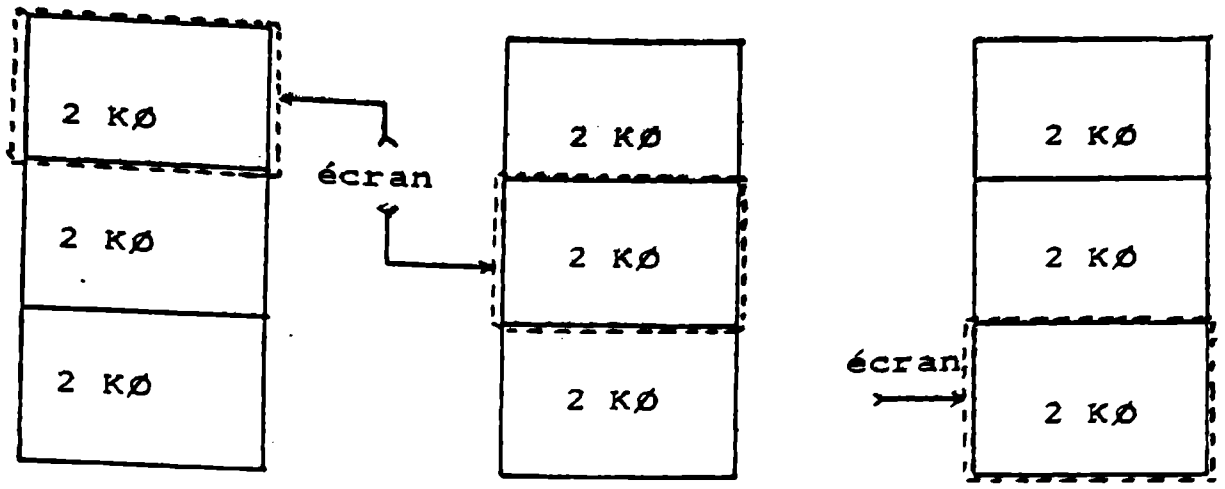
  La mémoire écran est découpée en partitions ou zones de travail. Chaque zone est considérée comme une consôle virtuelle. La fonction " multi-page " est donc le cas particulier de partitionnement de mémoire écran où chaque partition est une page.

- Partitionnement de l'écran :

  Indépendamment de la mémoire, l'affichage sur l'écran peut être partitionné. Chaque partition écran est un sous-ensemble d'une partition mémoire qui constitue une fenêtre visualisée. Les partitions bien que juxtaposées sur l'écran sont totalement indépendantes les unes des autres ; une seule est accessible à un instant donné et le curseur ne peut sortir de l'espace écran correspondant.

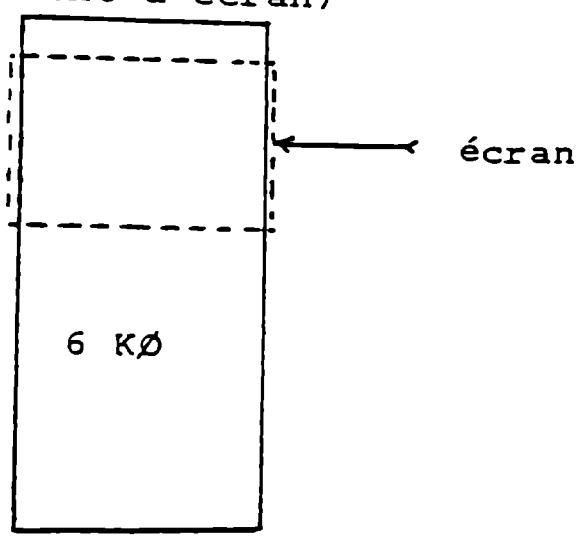
Exemples d'applications avec 3 X 2 KØ de mémoire écran :

- Multi-page : 3 partitions mémoire de 1 page : pas de partition d'écran.

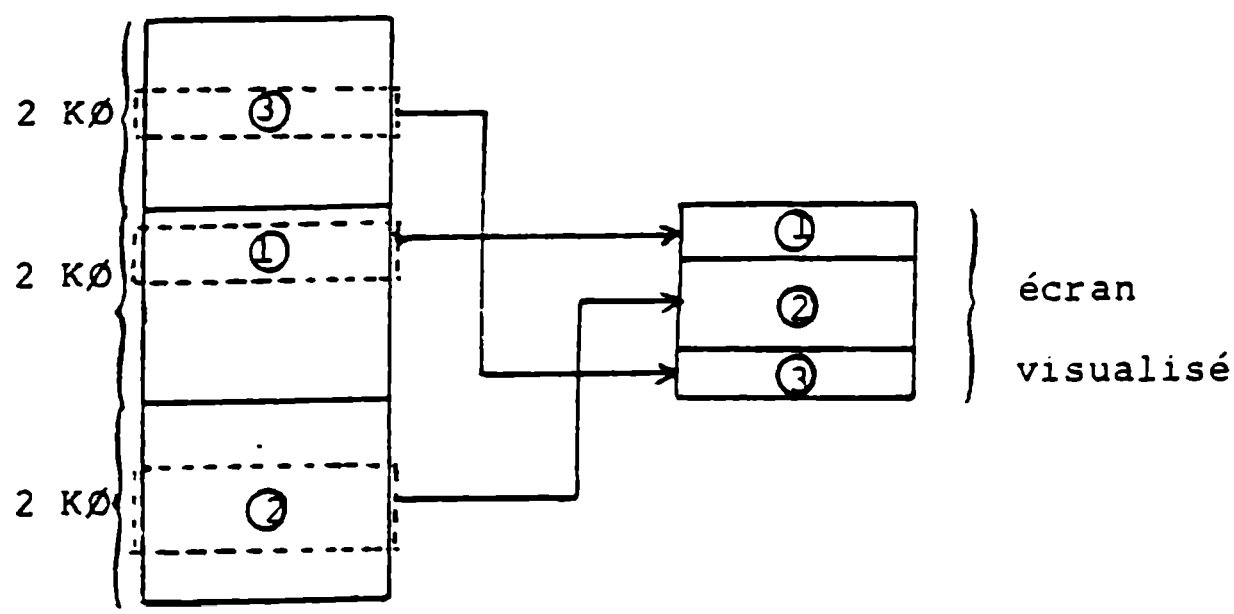


lère page visualisée 2ème page 3ème page

- écran mobile sur 1 partition unique de la mémoire (sans partitionnement d'écran)



- écran partitionné : 3 partitions de mémoire de 1 page
  3 partitions de l'écran



Commandes :

Trois commandes sont nécessaires à ces fonctions:

- Une commande de définition des partitions de la mémoire d'écran
- Une commande de sélection d'une partition de la mémoire d'écran où les données sont introduites ou extraites et où le curseur, unique, peut pointer
- Une commande de définition de partitionnement de l'écran.

Il est évident que ce type de fonctions agit directement sur l'architecture de la logique de commande puisqu'il faut gérer un espace mémoire écran différent.

Conclusion

L'étude des fonctions nouvelles du terminal fait apparaitre une évolution générale du terminal vers :

- Une augmentation du nombre d'interfaces vers des appareils périphériques qui sont optionnels
- Une amélioration des fonctions visuelles
- Un accroissement des fonctions logicielles.

LOGIQUE DE COMMANDE

SOLUTIONS POSSIBLES ET CHOIX

La logique de commande comprend deux ensembles :

- La logique de base
- La logique des interfaces d'appareils périphériques optionnels.

Les circuits de la logique de base sont sur un circuit imprimé (carte de base) et assument toutes les fonctions principales. Ceux de la logique des interfaces sont assemblés sur des cartes d'interfaces, chacune d'elle étant spécifique d'un type d'appareil périphérique.

Ce découpage physique a été réalisé pour des raisons économiques, étant donné le caractère optionnel des appareils périphériques. Un poste de travail sans imprimante, par exemple, peut être équipé d'un terminal sans la carte d'interface ' imprimante '.

Parmi les problèmes de choix posés par l'étude, les plus intéressant sont développés.

- Choix d'organisation de la logique des cartes d'interfaces.
- Choix du type de visualisation
- Choix des types de mémoires
- Choix du type de contrôleur d'accès direct à la mémoire.

1. PRINCIPE DE L'ARCHITECTURE DE LA LOGIQUE DE COMMANDE

La logique de commande du terminal est réalisée à base de circuits à très grande intégration (circuits LSI) de type MØS.
Les composants LSI disponibles sur le marché vont en partie déterminer le type d'architecture.

La logique de commande constitue à elle seule un système avec :

- Une unité centrale
- Une mémoire
- Des périphériques

Les échanges entre ces ensembles sont organisés à partir de trois types de liaisons :

- Le bus de données
- Le bus d'adresses
- Le bus de contrôles.

Les éléments de la logique dont le choix n'a pas fait l'objet d'études nouvelles, par rapport aux générations antérieures de terminaux de visualisation, seront décrits de façon succincte.

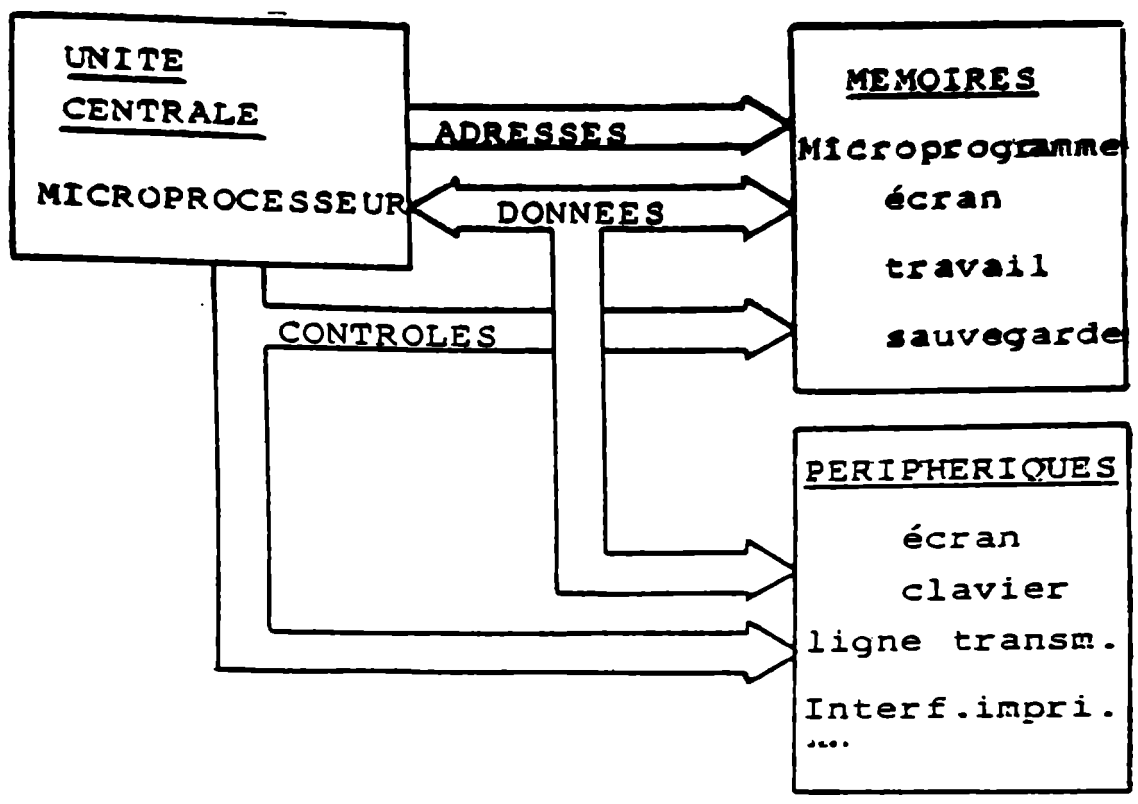


Figure 4 : ARCHITECTURE GENERALE

1.1. L'unité centrale : le microprocesseur

C'est le microprocesseur 8085-A du fabricant Intel.

Il a été retenu pour :

- Ses caractèristiques fonctionnelles et électriques suffisamment performantes pour traiter les fonctions demandées.
- Son prix
- Sa disponibilité sur le marché des composants (multi-source et grandes quantités).

Ses caractèristiques principales sont :

- Une technologie NMOS compatible TTL.
- Une alimentation unique 5 volts.
- Un bus de données de 8 bits, compatible avec le format des codes en provenance de la ligne de transmission.
- Un temps de cycle d'horloge de 325 ns et un temps de cycle d'instruction de 1,3 µs minimum, compatibles avec les débits maximum sur la ligne (100 000 bps ⇒ 1 caractère toutes les 100 µs).
- Quatre niveaux d'interruptions utilisés par les circuits périphériques.
- Arithmétique décimale, binaire et double précision
- Possibilité d'adressage direct de 64 Kilo/mots de mémoire.

1.2. La mémoire

Quatre mémoires distinctes sont nécessaires à la logique de commande :

- La mémoire de commande
- La mémoire d'écran et le générateur de caractères
- Une mémoire de travail
- Une mémoire de sauvegarde.

1.2.1. La mémoire de commande

Pour piloter la logique de commande,des informations sont rangées dans la mémoire de commande (microprogramme).

L'appareil peut remplacer l'un quelconque des quatre terminaux précédemment cités avec le même matériel, mais la mémoire de commande reste spécifique d'un type de fonctionnement.

Pendant le fonctionnement, cette mémoire est lue uniquement et est donc de type mort, excepté pour l'émulation d'un des terminaux (VU3) dont le microprogramme doit être chargeable à partir d'un contrôleur de terminaux.

1.2.2. La mémoire d'écran et le générateur de caractères

- Les codes à visualiser sur l'écran sont rangés dans une mémoire d'écran qui est écrite lorsqu'un caractère ou un attribut visuel sont reçus et qui est lue pour le rafraîchissement de l'image.

- Les codes de la mémoire d'écran doivent être transcodés par un générateur de caractères (mémoire morte) pour être exploitable au niveau du signal de visualisation.

1.2.3. La mémoire de travail

Pour stocker provisoirement des informations avant le traitement, pour exécuter des opérations logiques transitoires, etc..., le microprogramme a besoin d'une mémoire vive de travail.

1.2.4. La mémoire de sauvegarde

Lors d'une coupure secteur ou d'une mise hors tension de l'appareil, certaines informations variables (paramètres chargés par un opérateur) doivent être sauvegardés dans une mémoire vive dite mémoire de sauvegarde.

A l'exception de la mémoire du générateur de caractères et éventuellement de la mémoire écran, toutes sont directement reliées au bus d'adresses. La capacité d'adressage totale de 64 Kilo-mots fournie par le microprocesseur ne doit donc pas être dépassée.

1.3. Les circuits périphériques

Les uns sont implantés sur la carte de base et les autres sur les cartes d'interfaces, mais ils sont tous vus de façon identique par le microprocesseur.

Le choix de certains d'entre eux n'a pas fait l'objet d'études approfondies, parce que leurs fonctions sont pratiquement standardisées et que les différents constructeurs offrent des circuits équivalents. Dans ce cas les circuits ' Intel ' ont été retenus afin d'assurer un maximum de compatibilité avec le microprocesseur.

- Le circuit d'interface de communication (8251-A Intel) ou USART réalise en partie la gestion de l'émission ou de la réception de données en mode synchrone ou asynchrone (sérialisation/désérialisation, parité, contrôle de débit, contrôle d'erreurs etc...).

- Le circuit d'interface clavier (8255-A Intel) ou PPI verrouille les données en provenance du clavier. C'est ensuite le microprocesseur qui vient les prélever par scrutation périodique; la fréquence de scrutation étant supérieure à la cadence de transfert des codes du clavier.

  Le circuit permet aussi l'envoi de commandes vers le clavier (voyants lumineux, alarme,...).

- Le circuit de compteurs (8253-5 Intel) ou PIT peut,entre autre, être utilisé comme générateur de vitesses de transmissions pour l'USART et comme générateur d'horloges (horloge de fréquence d'alarme sonore par exemple).

1.4. Les bus d'interconnexions

Le bus de données de 8 bits relie tous les circuits programmables. Deux circuits, un émetteur, l'autre récepteur, utilisent simultanément le bus; tous les autres doivent être dans l'état de sortie haute impédance. Au niveau du microprocesseur, le bus de données (AD0-AD7) est commun au bits de poids faibles du bus d'adresse.

C'est pendant des temps de cycles bien définis que chaque bus est validé. Le bus d'adresses de 16 bits relie le microprocesseur aux mémoires et également à une logique de décodage nécessaire à la sélection des circuits mémoires et périphériques.

Le bus de contrôle sert de support à toutes les fonctions de contrôle du microprocesseur : lecture et écriture de mémoires ou de périphériques, interruptions, attentes, etc...

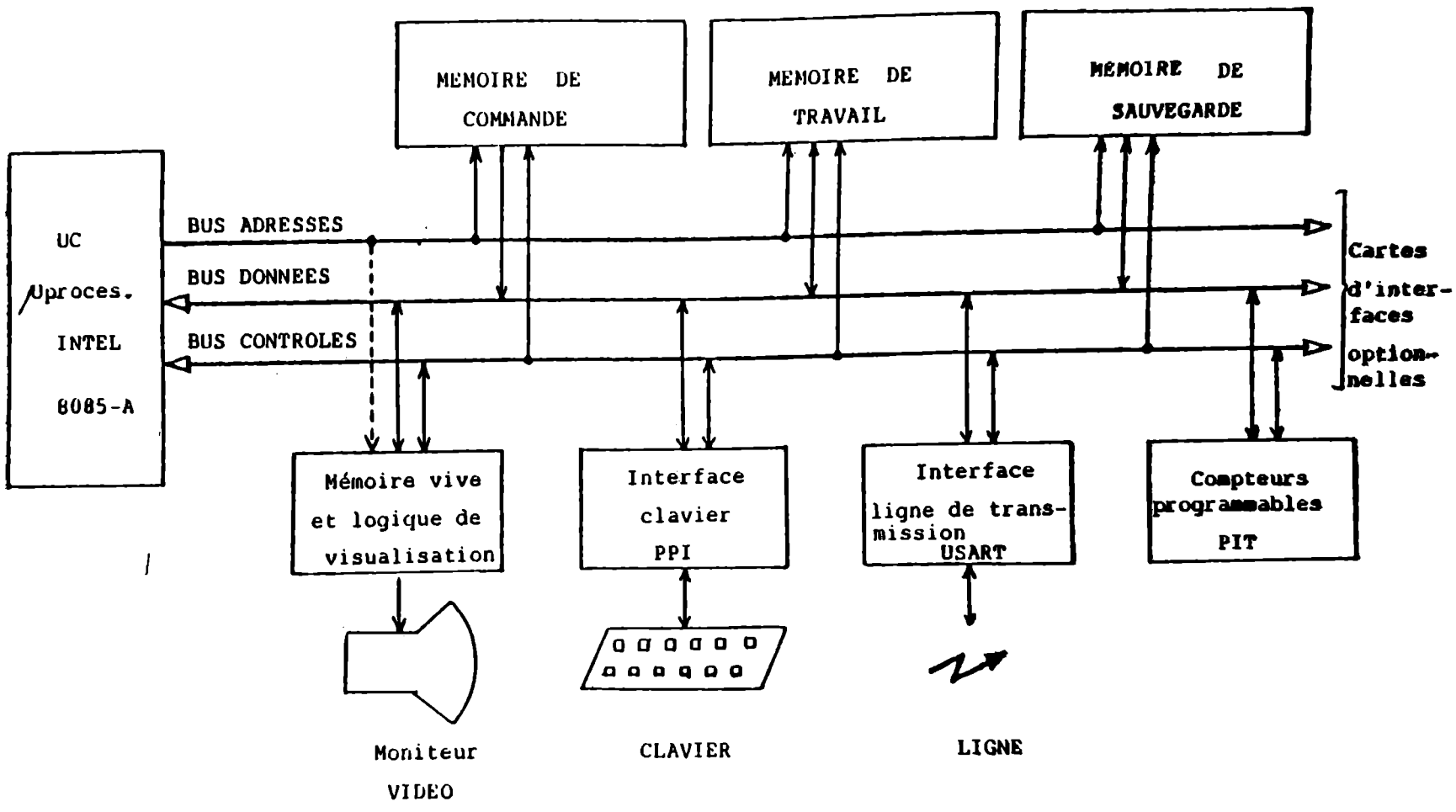


Figure 5 : ARCHITECTURE PAR ENSEMBLES LOGIQUES

2. LOGIQUE DES INTERFACES OPTIONNELS

Le terminal est conçu de façon telle que, à la logique de base, il est possible d'associer des logiques supplémentaires réalisées sur des cartes (circuits imprimés) dites ' options ' qui permettent la connexion d'un appareil périphérique : interface imprimante, interface lecteur de badges, interface lecteur de documents, interface photostyle, etc...

2.1. Principes de fonctionnement

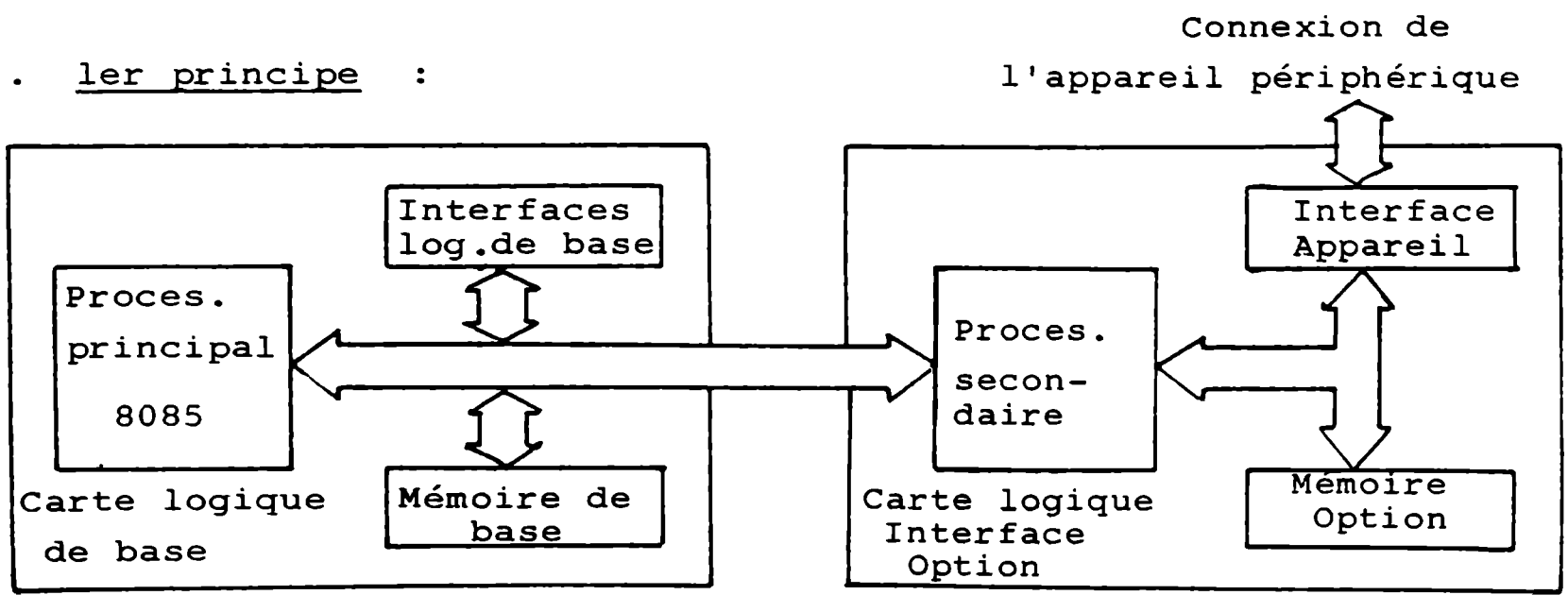
L'interface permet d'adapter le dialogue de l'appareil périphérique, récepteur ou émetteur, à celui du terminal.
La logique de commande réalise principalement des fonctions de stockage (mémoires tampons), de transcodage (mise en forme des messages émis, décodage des messages reçus)et de transmission (sérialisation/désérialisation, vitesses,...) de l'information.

Pour exécuter ces fonctions chaque carte option dispose d'un espace mémoire de commande.

La logique de commande peut être conçue suivant deux principes différents :

- L'interface autonome est gérée par son propre circuit programmable.
- L'interface est gérée par le microprocesseur de la carte principale comme un circuit périphérique.

. ler principe :



La technique consiste à développer un système secondaire à partir d'un processeur qui reste esclave du microprocesseur de la logique de base, mais qui a son propre jeu d'instructions, une mémoire de commande, une mémoire de travail ainsi que des bus d'entrées/sorties.

Il existe sur le marché des circuits adaptés aux fonctions demandées, ce sont :

Les circuits d'interface périphérique universel (UPI) du type 8741A ou 8041A de chez INTEL.

Les circuits microprocesseurs ' bon marché ' aux fonctions moins évoluées que celles des microprocesseurs type 8085A ; exemple : le 8048 ou le 8748 de chez INTEL.

- Les circuits 'UPI' intègrent dans le même boitier :

  . Une unité centrale (8 bits)

  . 64 octets de mémoire vive RAM (8 bits),

  . 1 Kilo-octet de mémoire morte (ROM ⟶ 8041 ou EPROM ⟶ 8741).

  . Des registres d'interface BUS de données (8 bits) et deux ports d'entrée/sorties (8 bits).

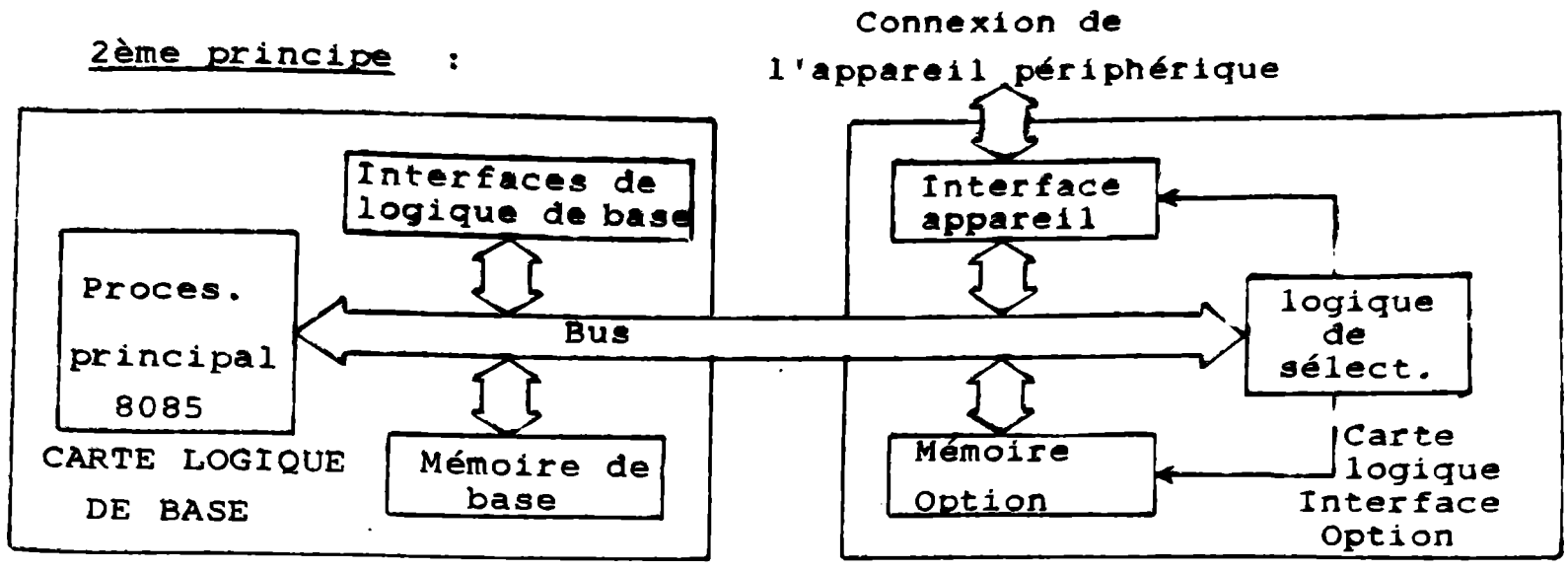
  . Un sytème d'horloge fonctionnant jusqu'à 6 MHz et un compteur d'évènements.

  Le circuit est entièrement programmable grâce à son jeu d'instructions et constitue à lui seul un micro-système dont la souplesse de programmation le rend adaptable à plusieurs types d'interfaces.

- Les circuits de type 80/8748 présentent les mêmes possibilités que le UPI, plus une facilité d'extension de la mémoire.

Avec de tels circuits le microprocesseur de base n'intervient, dans le dialogue terminal/appareil périphérique ou ligne de transmission/appareil périphérique, que pour effectuer des transferts de données.

Lorsque l'appareil est un émetteur (cas des lecteurs de badges par exemple), le processeur d'interface émet une demande d'interruption du microprocesseur principal à la fin du traitement du message, afin que le transfert puisse s'effectuer soit vers l'écran soit vers la ligne de transmission .

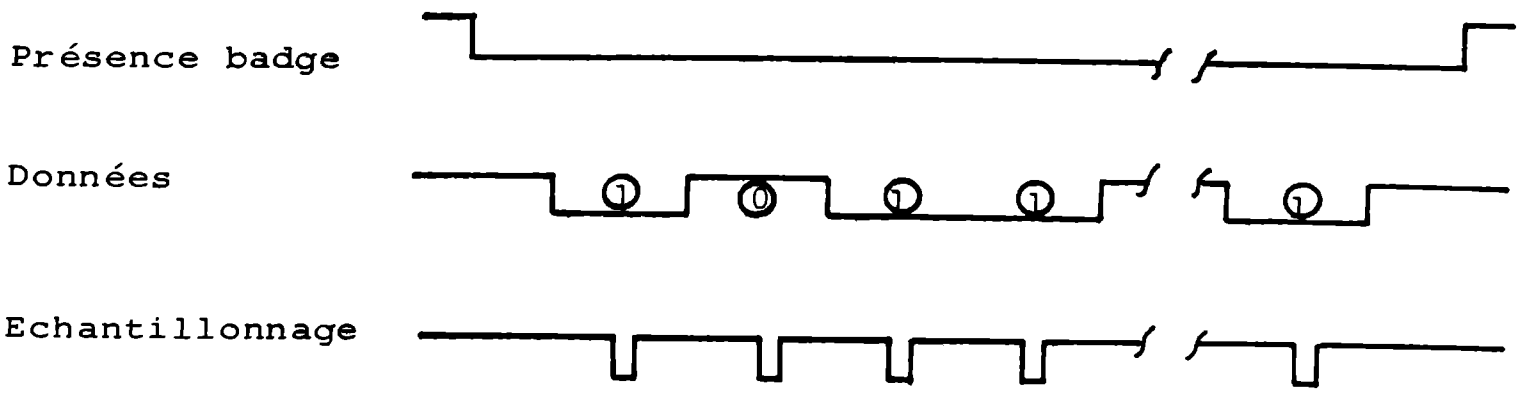


Le principe consiste à considérer les interfaces des appareils périphériques de la même façon que les interfaces de la logique de base. Dans ce cas c'est le microprocesseur 8085 qui contrôle et gère le fonctionnement des interfaces optionnels. La mémoire de commande, et éventuellement la mémoire ' tampon ', nécessaire au fonctionnement de l'interface sont implantées sur la carte option.

La transmission vers ou en provenance de l'appareil connecté s'effectue par un outil d'interface adapté. Le circuit d'interface de transmission, USART par exemple, est couramment employé pour assurer l'interface avec une imprimante.

2.2. Application à l'interface ' lecteur de badge ' :

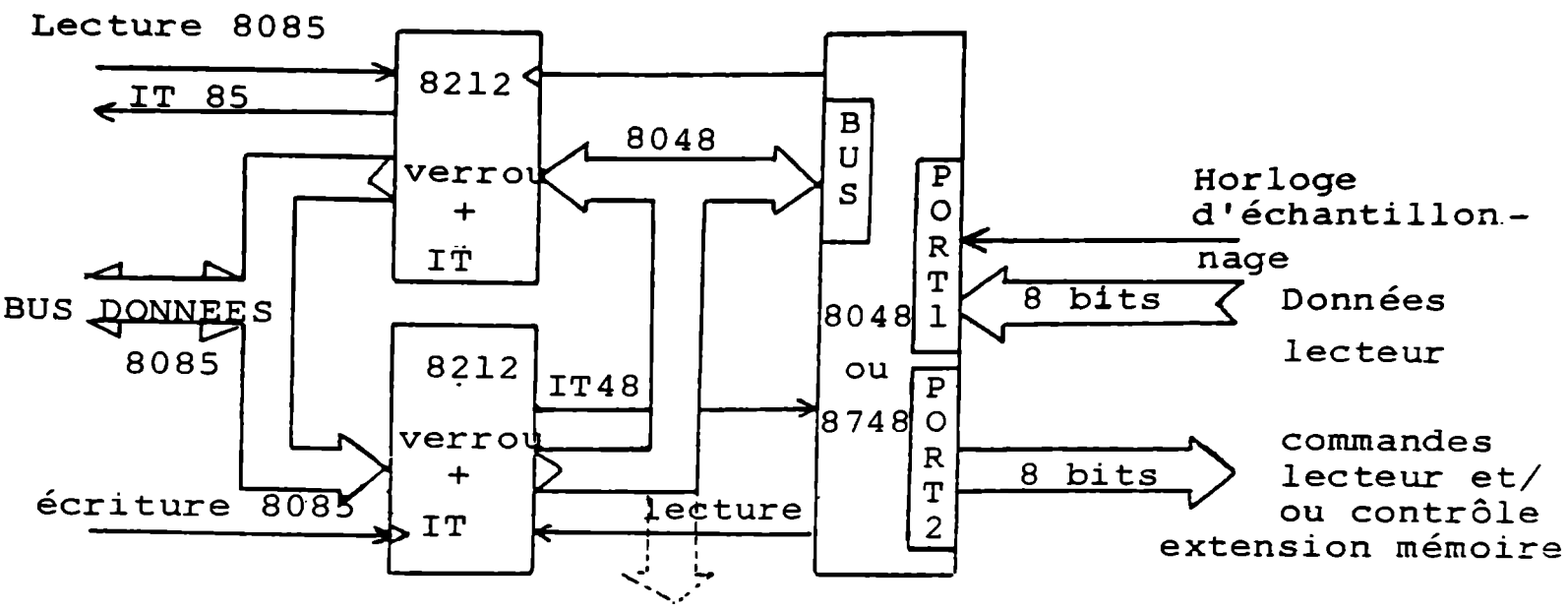
Le lecteur de badge est un appareil périphérique émetteur. Lorsqu'un badge y est lu, le message magnétique qu'il contient est directement envoyé vers le terminal sous forme d'impulsions électriques compatibles avec les technologies de circuits intégrés. Les lecteurs actuels transmettent l'information en série. La liaison fournit trois signaux : le signal d'information, le signal d'échantillonnage des données et le signal de présence d'un badge. L'ordre de grandeur de la période d'échantillonnage est de 100 µs.



lère solution pour réaliser l'interface lecteur de badge :

Un circuit de type 80/8748 est préféré au type UPI 80/8741 car :

- Il offre davantage de signaux de contrôle et la possibilité d'étendre sa capacité mémoire interne.
- Le UPI est un circuit commercialement nouveau ; il est mono-source et son prix est plus élevé que celui du 80/8748.



SYNOPTIQUE DE LA LOGIQUE INTERFACE LECTEUR DE BADGE

Un étage de mémorisation (verrou) est nécessaire afin de stocker temporairement les informations lorsque le dialogue entre les deux microprocesseurs n'est pas prêt.

Le message reçu sur le " port 1 " est stocké en mémoire vive du 80/8748. Sur reconnaissance d'un code de fin de message, le microprocesseur traite ce message (mise en forme, contrôle de parité, message d'erreur) à l'aide de sa propre mémoire de commande.

Lorsque le message a été traité et accepté, le 80/8748 charge octet par octet les données du message dans l'étage de verrous.

Chaque octet chargé, déclenche une demande d'interruption du microprocesseur 8085 qui viendra le prélever.

Le synoptique proposé permet en outre la réception d'un message en liaison parallèle et autorise l'envoi de commandes à partir du 8085 ou du 8048 vers l'appareil périphérique (port2=8 bits). En tenant compte également de l'extension possible de la mémoire du 8048, cette logique permet de traiter, avec le même matériel, soit un interface lecteur de badge, soit un interface lecteur de document, ou un autre lecteur à transmission lente.

Les schémas détaillés de la logique sont fournis en annexe.

. <u>2ème solution pour réaliser l'interface lecteur de badge</u> :

Il n'existe pas sur le marché de circuit intégré évolué spécifiquement adapté à l'interface d'un lecteur de badge ; toutefois, les fonctions à réaliser relativement simples sont :

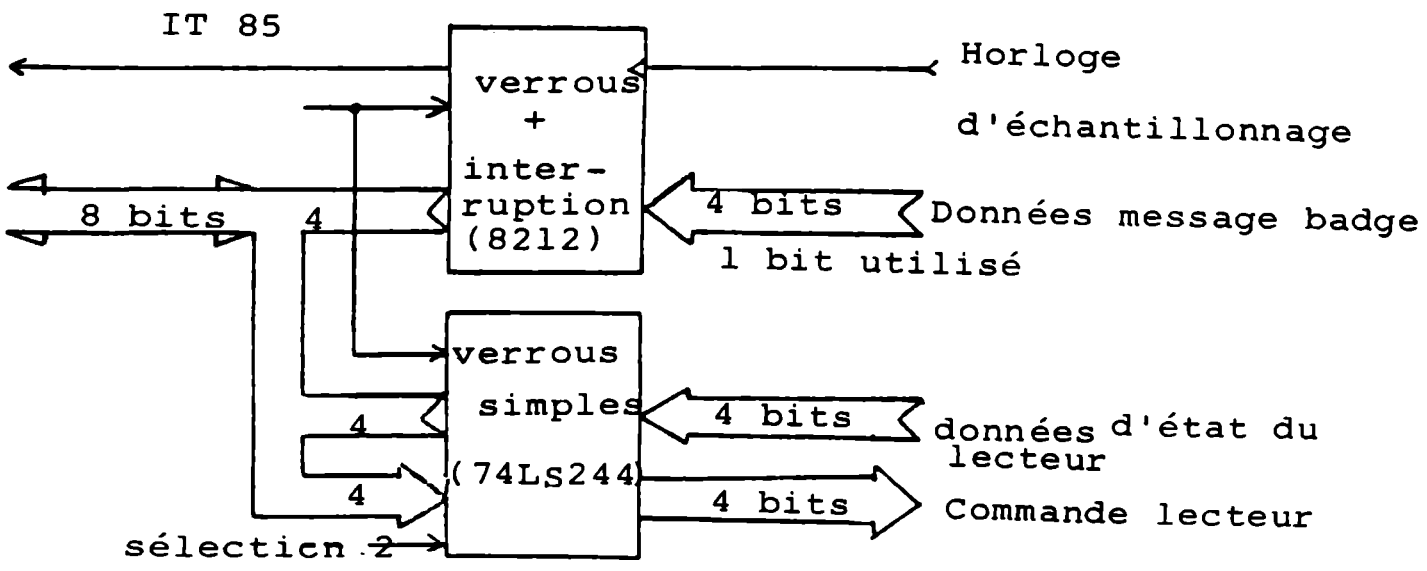
- La mémorisation des bits d'information reçus ou des bits de commande à émettre vers le lecteur.
- La génération d'une interruption du 8085 au moment de l'échantillonnage des bits du message

Pour cela plusieurs circuits sont possibles :

. Verrous avec interruption, type 8212.

. Circuit d'interface de transmission série, type 8251.

Le principe de fonctionnement étant le même quel que soit le circuit, seule la réalisation avec verrous est décrite.



-<u>SYNOPTIQUE DE LA LOGIQUE D'INTERFACE LECTEUR DE BADGE</u>

Sur le signal d'échantillonnage, le signal de données du message est mémorisé dans le 8212 et l'interruption est demandée (IT 85). Lorsque le microprocesseur est disponible il prélève le bit mémorisé et le traite.

Cette réalisation permet la réception d'un message en liaison parallèle et autorise l'envoi de commandes vers le lecteur. La même logique permet de gérer l'interface avec les lecteurs de badges et de documents utilisés dans la compagnie ; et avec d'autres appareils, types lecteurs lents. Les schémas détaillés de la logique sont fournis en annexe.

2.3. Comparaison des solutions et choix :

2.3.1. Inconvénients relatifs à chaque principe :

1ère SOLUTION (8048)	2ème SOLUTION (circuit d'interface)
. L'espace de mémoire dont dispose le microprocesseur secondaire n'est pas assez important pour gèrer des interfaces complexes (impressions de type évolué par exemple). . La lenteur d'exécution des instructions ($\geq$ 2,5 µs) pose des problèmes pour des transmissions rapides (imprimantes à 9600 bps)	. L'utilisation de circuits spécialisés implique pratiquement le développement d'une carte optionnelle par interface. . L'espace de mémoire nécessaire aux options, est pris sur l'espace mémoire du microprocesseur 8085. . Le microprocesseur est occupé pendant tout le temps de traitement des fonctions d'interface optionnelles.

2.3.2. Avantages relatifs à chaque principe :

1ère SOLUTION (8048)	2ème SOLUTION (circuit d'interface)
. Possibilité d'utiliser le même matériel pour réaliser tous ou plusieurs types d'interfaces, c'est le programme de commande implanté (EPROM) qui caractérise le type d'interface. . Pas d'occupation de l'espace mémoire et peu d'occupation du temps du microprocesseur 8085.	. La spécialisation de l'interface réduit le nombre de es fonctions, et par conséquent simplifie le matériel. . La conception du programme de commande global (base + options) est simple ; chaque option est vue par le microprocesseur comme un interface de la logique base. . Le coût surtout est plus faible (12,5 % plus faible pour l'interface lecteur de badge).

Dans le cadre de l'étude d'un terminal nouveau, la première solution est intéressante, car elle offre une meilleure couverture pour la réalisation de l'interface d'appareils périphériques futurs (peu d'incidences sur le microprogramme de base puisque la gestion de l'interface est indépendante). Toutefois les débits de transmisssions des appareils périphériques doivent rester faibles.

Dans le cadre de l'étude présente, le second principe est choisi car, pour émuler les appareils existants dont les fonctionnements sont variés (traitement de l'imprimante par exemple), la spécification des cartes d'interfaces rend plus simple le matériel de chaque carte et le programme de commande qui lui est associé.

3. CHOIX DE LA LOGIQUE DE VISUALISATION :

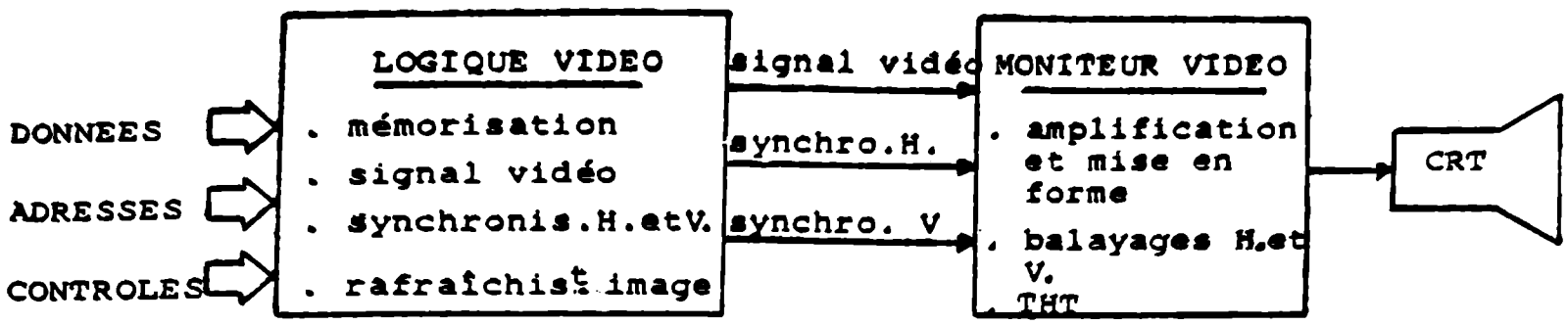


figure 6 : SYNOPTIQUE FONCTIONNEL

La logique de visualisation réalise l'interface entre les bus du microprocesseur et le moniteur vidéo. Le moniteur transforme les signaux numériques en signaux analogiques exploitables par le tube

Les codes à visualiser sont rangés dans une mémoire dite ' mémoire d'écran '. Le signal vidéo est réalisé à partir des codes de la mémoire, d'un transcodeur - le générateur de caractères -, d'un sérialisateur et d'une horloge de comptage de points vidéo.

Des compteurs fournissent les synchronisations, horizontales et verticales, nécessaires au balayage du spot sur le tube. L'écran est balayé horizontalement. Mais la fonction principale de la logique vidéo reste le rafraîchissement d'image qui peut être réalisé suivant différents principes.

3.1. Principes de gestion du rafraîchissement de l'image

Le choix du principe de gestion du rafraîchissement, c'est à dire de la lecture de la mémoire d'écran et de l'envoi des données sur le tube 50 fois par seconde, a de l'influence à la fois sur la qualité de l'image, sur les performances du terminal et sur le coût du matériel. Il est donc important au cours de l'étude, d'envisager chaque possibilité. Les deux principes fondamentaux sont :

- Le rafraîchissement à mémoire d'écran dite 'spécialisée'.
- Le rafraîchissement à mémoire d'écran dite 'banalisée'.

3.1.1. <u>Rappel</u> (voir figures 7 et 8)

Le balayage du faisceau du tube décrit sur l'écran une suite de lignes horizontales. L'apparition des caractères est reconstituée par excitation du phosphore sur l'écran en des points précis et sur plusieurs lignes consécutives (n). L'alignement des caractères, visualisés sur n lignes, constitue une rangée de X caractères possibles. L'écran peut contenir Y rangées de caractères.

Le signal vidéo qui code l'excitation du faisceau lumineux est donc constitué d'une succession de codage de segments de caractères (C1, C2,...).

La mémoire d'écran contenant des codes de caractères entiers, il est nécessaire d'utiliser un transcodeur (générateur de caractères) pour générer le code segment de caractères correspondant à un caractère et un numéro de ligne parmi n.

Pour l'appareil étudié, de 2000 caractères affichables comprenant 15 lignes et dont le rafraîchissement s'effectue à 50 Hz, le générateur de caractères reçoit une nouvelle adresse toutes les $\frac{20.10^{-3}}{2000 \times 15}$ = 666 ns. Sachant que le transfert d'un caractère entre deux mémoires à partir du microprocesseur exige 28 cycles d'horloge(*), c'est à dire 9,1 µs, il est évident que le rafraîchissement ne peut pas être géré par le microprocesseur.

(*) Instructions 8085 :

MOV M,R (7 cycles) + XCHG (4) + MOV r, M (7) + XCHG (4) + INX (6).

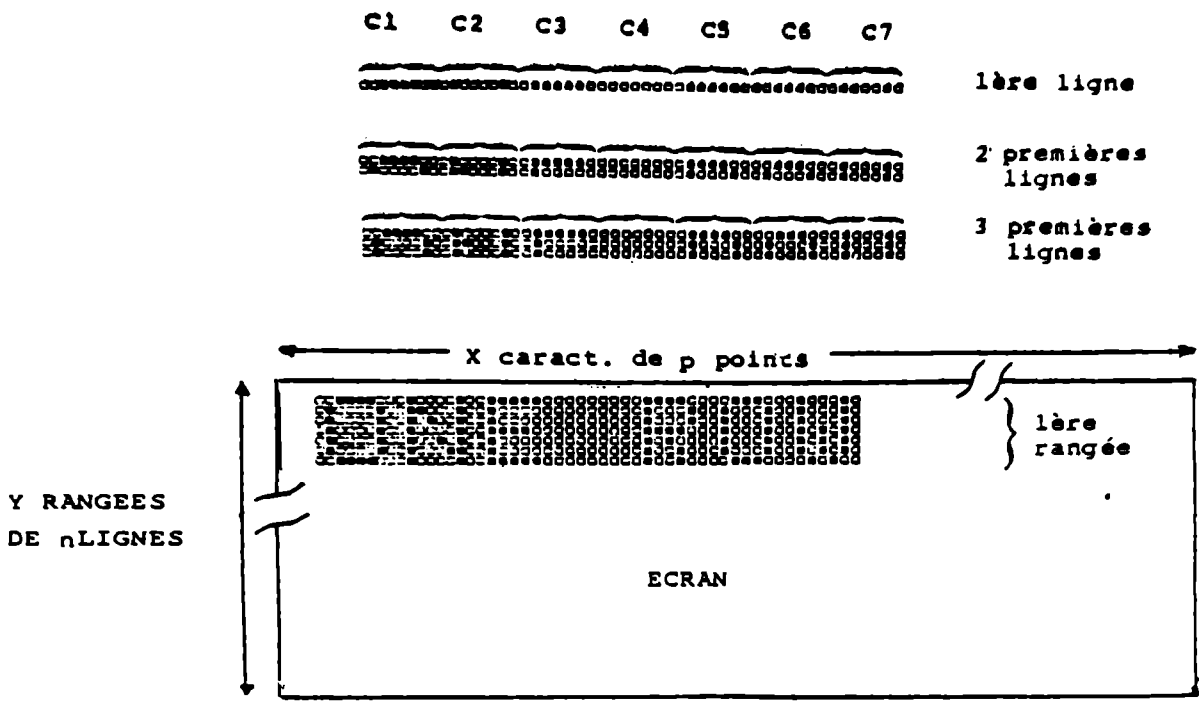


Figure 7 : AFFICHAGE DES CARACTERES

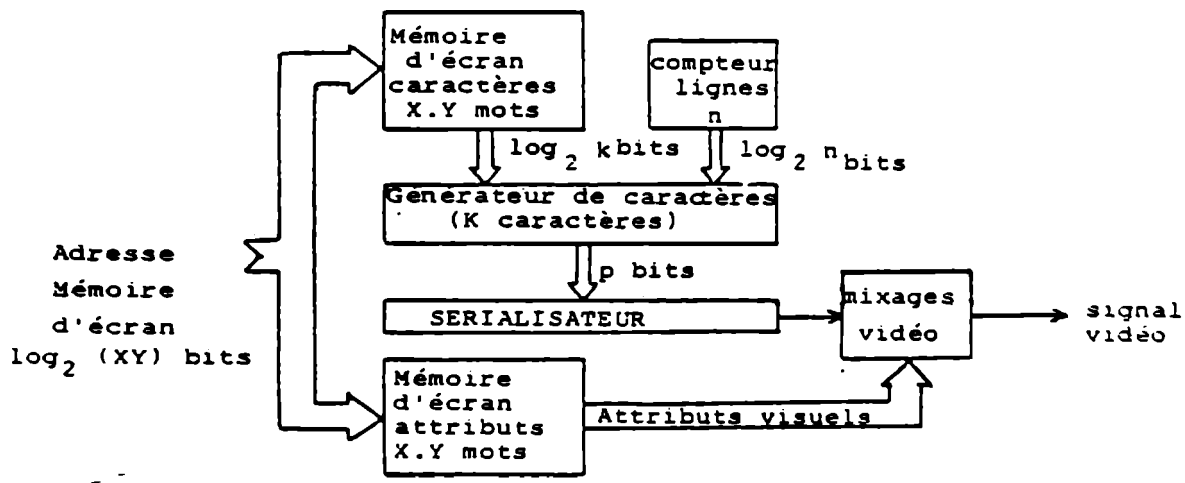


Figure 8 : LOGIQUE DE RAFRAICHISSEMENT

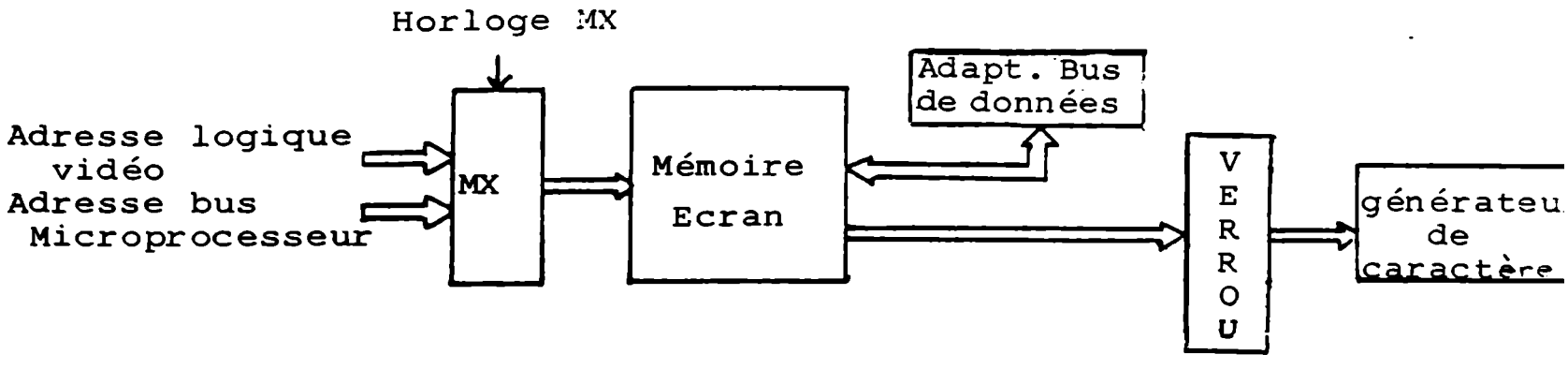
3.1.2. Premier principe : mémoire d'écran spécialisée (fig. 9)

La mémoire d'écran est dite spécialisée car elle se différencie des autres mémoires du système par son adressage indépendant. Pour exécuter la fonction de rafraîchissement, les adresses sont en effet données soit par un contrôleur de TRC soit par une logique de comptage. Il n'y a pas d'utilisation du microprocesseur ni de ses bus.

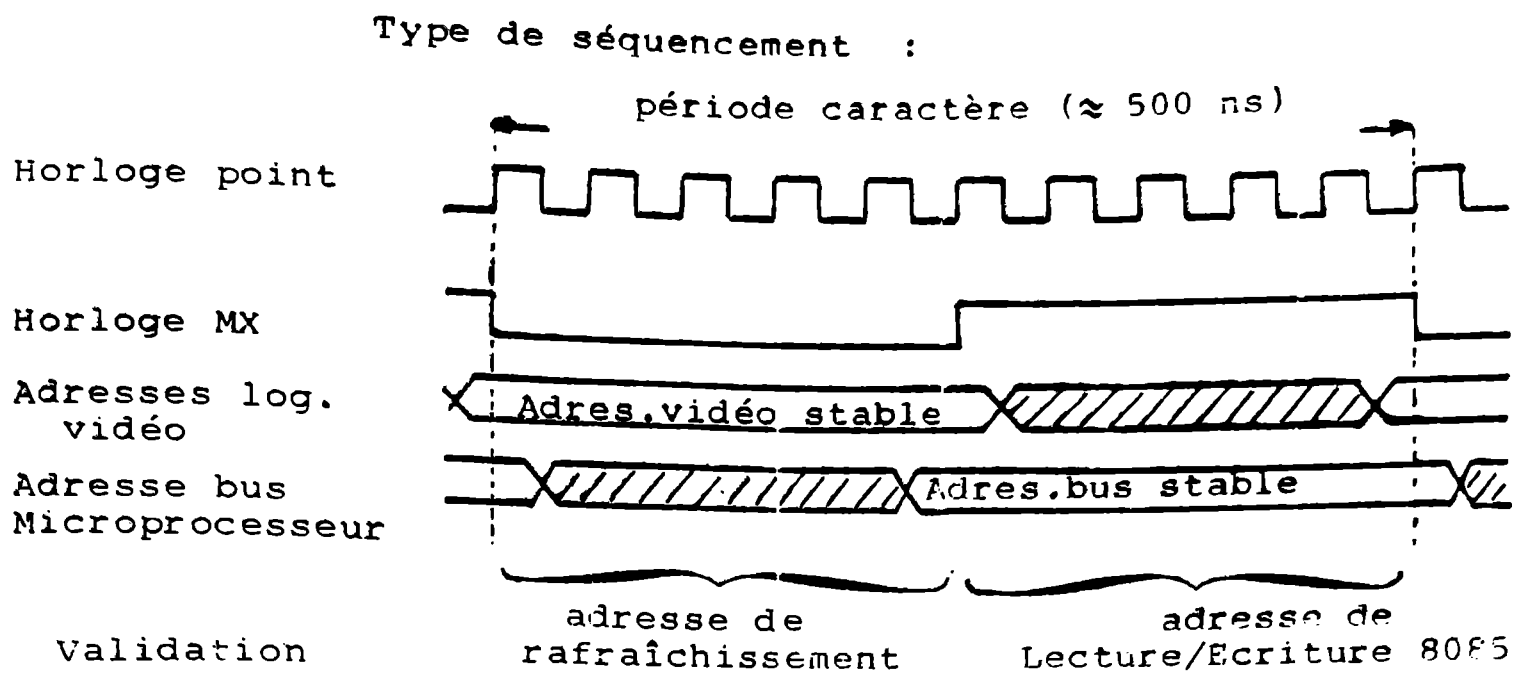
Toutefois le microprocesseur peut accèder à la mémoire d'écran pour y écrire de nouveaux caractères ou pour en lire le contenu.

- La solution la plus simple, avec laquelle le microprocesseur est prioritaire, consiste à interrompre l'accès mémoire pour le rafraîchissement d'image lorsque le microprocesseur doit effectuer des lectures ou écritures dans cette mémoire ; mais à chacune de ces interventions, l'interruption du rafraîchissement d'image provoque une détérioration de la qualité de l'image (phénomène flagrant pendant une écriture ou une lecture d'un bloc de caractères).

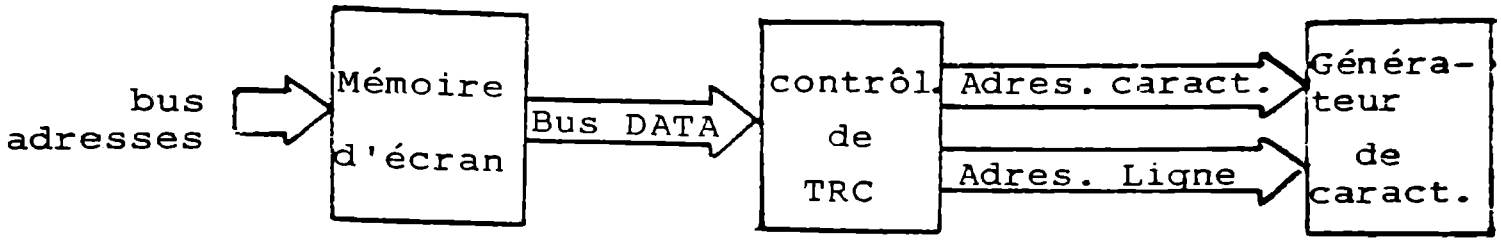
- Une autre solution, pour assurer le multiplexage de l'accès à la mémoire écran entre la logique vidéo et le microprocesseur, consiste à réaliser une logique d'horloges. Cette logique est indépendante du microprocesseur et génère systématiquement le multiplexage de l'accès mémoire.



A chaque cycle d'écriture ou de lecture à partir du microprocesseur l'accès par les bus à la mémoire n'est validé qu'à l'intérieur d'un créneau. La logique de visualisation est synchronisée sur l'horloge point (oscillateur), et il est donc possible de fournir les adresses de rafraîchissement au bon moment par rapport à l'horloge 'MX'. Par contre le rythme du microprocesseur est asynchrone par rapport à la logique de visualisation et il est nécessaire de forcer ce dernier dans un état d'attente (1 cycle d'attente du 8085 = 320 ns) tant qu'un cycle d'accès bus ne s'est pas écoulé. Dans cette solution la logique vidéo est donc prioritaire par rapport au microprocesseur.



3.1.3. Deuxième principe : mémoire d'écran banalisée (fig. 9)



La mémoire d'écran est dite banalisée car elle est reliée au bus d'adresses et de données comme une autre mémoire du système. Dans ce cas c'est à un processeur de gèrer le transfert des données vers la logique vidéo pour assurer le rafraîchissement d'image.

Il a été montré que l'adressage directe du générateur de caractère toutes les 666ns n'est pas compatible avec la vitesse de travail du microprocesseur (1 transfert toutes les 9,1 µs) ou d'un circuit spécialisé type DMA (1 transfert toutes les 1 µs). Pour cette raison, ce type de rafraîchissement nécessite l'emploi d'un circuit contrôleur de CRT, intercalé entre la mémoire d'écran et le générateur de caractères, qui permet de mémoriser dans un FIFO une rangée entière de caractères.

Le contrôleur se charge de fournir 15 fois les 80 codes des caractères d'une rangée, en incrémentant à chaque fois l'adresse ligne. Ce fontionnement divise donc par 15 le nombre de transferts à effectuer entre la mémoire d'écran et la logique vidéo.

La période de ces transferts devient $\frac{20.10^{-3}}{2000}$ = 10 µs pour 2000 caractères et $\frac{20.10^{-3}}{4000}$ = 5 µs pour 2000 caractères et 2000 attributs visuels associés. Le transfert ne peut toujours pas être réalisé par le microprocesseur, mais il devient possible de l'éxécuter à partir d'un circuit contrôleur d'accès direct en mémoire DMA. Ce circuit se substitue au microprocesseur pour transférer dans un temps bien plus court des blocs de données à adresses contigües entre une mémoire (mémoire d'écran) et un périphérique (contrôleur de CRT).

Le DMA est initialisé par le microprocesseur avec une adresse de début de transfert (adresse du premier caractère d'une rangée d'écran) et avec un compte d'octets à transférer (Y x 80 caractères); puis il prend le contrôle des bus et exécute un transfert des Yx80 caractères. Le transfert d'un caractère s'exécute en 3 ou 4 périodes d'horloge (3 x 325 ns = 1 µs ou 4 x 325 ns = 1,3 µs) suivant le type de DMA.

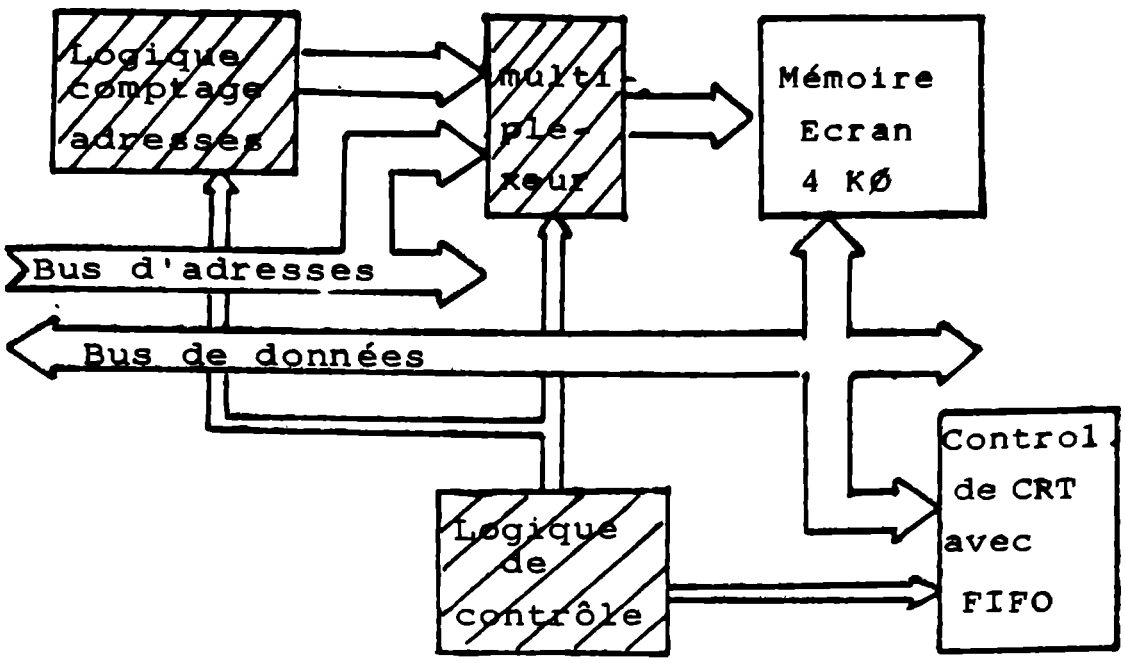
Le temps de transfert sur les bus pour une image entière est donc pour 2000 caractères et 400 attributs visuels (*) de :

2400x3x325 ns = 2,34 ms à 2400x4x325 ns = 3,12 ms ce qui représente pour une image toutes les 20 ms (50 Hz) un temps d'occupation du bus de 11,7 % à 15,6 %.

(*) Le contrôleur de CRT actuellement adapté à ce type de fonctionnement comprend un FIFO de 80 caractères et un FIFO de 16 attributs visuels seulement ce qui autorise 16x(25 rangées) = 400 attributs par écran contrairement aux 2000 souhaités, ce problème sera abordé ultérieurement.

3.1.4. Autre solution (pour mémoire) :

Il faut signaler une autre possibilité de rafraîchisssement, qui est un compromis des deux principes précédents puisque la mémoire d'écran est adressée par un bus spécialisé, mais est reliée en sortie au bus de données du microprocesseur.



L'intérêt de cette solution consiste à exploiter les intervalles de temps pendant lesquels le microprocesseur n'utilise jamais le bus de données pour transférer les données de rafraîchissement de la mémoire écran vers le contrôleur de TRC. Ceci permet d'effectuer le rafraîchissement sans augmenter le temps d'occupation du bus.

Les microprocesseurs exécutent deux types de cycles machines : les cycles d'exécution des instructions (opérations internes, lectures/écritures en mémoires ou périphériques) et les cycles de recherche en mémoire programme des codes d'instructions (cycles FETCH). Ces cycles de recherche de 4 périodes d'horloge, n'utilisent pas le bus de données pendant la dernière période, soit 320 ns avec une 8085. Ce temps est suffisant pour exécuter une écriture dans le contrôleur de TRC.

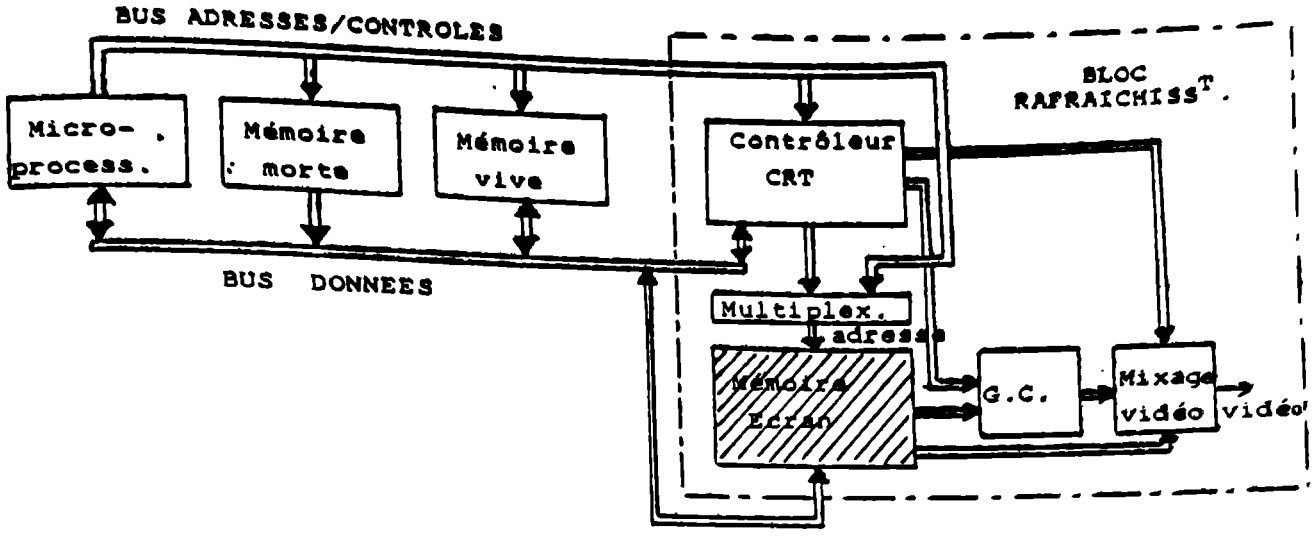
D'autre part, sachant qu'un cycle de recherche est lancé entre chaque exécution d'instruction et que l'exécution de l'instruction la plus longue dure 13 périodes d'horloge, soit 13x325 ns $\approx$ 4,3 µs, il est possible de transférer les 4000 mots de la mémoire d'écran vers le contrôleur dans un temps Tr tel que :

$$Tr \leqslant 4000 \times 4{,}3\ \mu s = 17{,}2\ ms < 20\ ms$$ (période de rafraîchissement d'image à 50 Hz (*)).

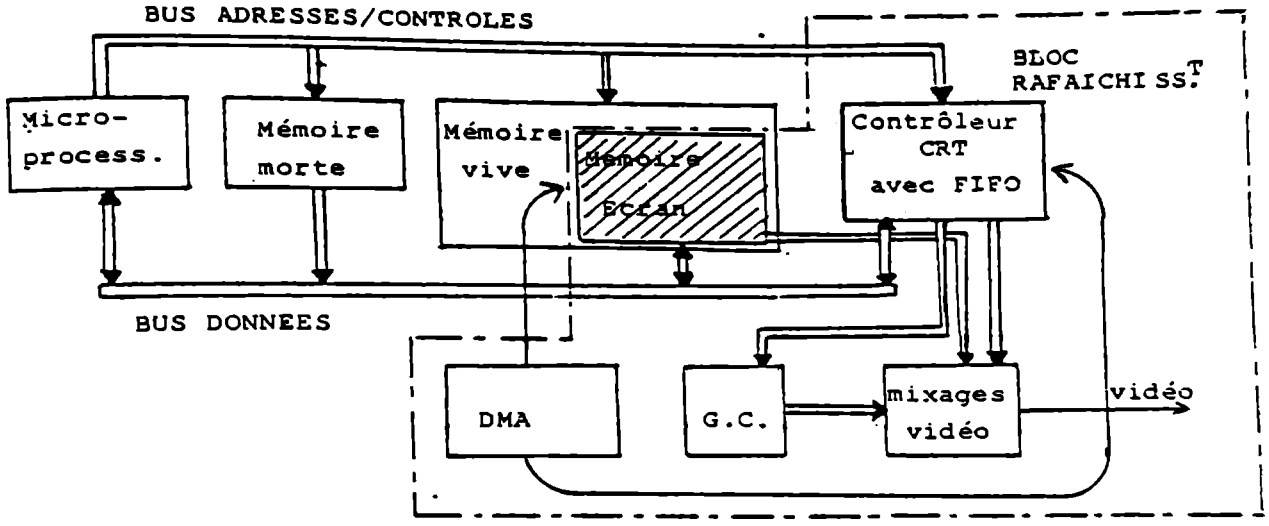
Intéressant pour ses performances, ce principe n'est pas retenu pour les raisons suivantes :

- La logique supplémentaire (hachurée sur la figure) est coûteuse comparée à l'amélioration des performances.

- Ce fonctionnement interdit l'occupation prolongée ou permanente du bus de données par un autre cicuit ou processeur tel que le contrôleur d'accès direct en mémoire (DMA); cette restriction limite dangereusement les ouvertures à de nouvelles fonctions.



MEMOIRE ECRAN SPECIALISEE



MEMOIRE ECRAN BANALISEE

Figure 9 : PRINCIPES DE RAFRAICHISSEMENT D'ECRAN

Pour le projet traité, cette solution interdirait le traitement des émissions/réceptions de données sur la ligne de transmission par un DMA.

(*) Pour une fréquence de 60 Hz (16 ms) ce fonctionnement ne peut pas être garanti 100 %.

3.2. Comparaison des deux principes (paragraphes 3.1.2. et 3.1.3.)

Le premier principe - à mémoire spécialisée - est envisagé avec un contrôleur de CRT de type 6845 de chez Motorola, qui fournit les adresses de la mémoire, et une logique de séquencement qui partage le temps d'accès mémoire entre les accès par le contrôleur de CRT et les accès par le microprocesseur.

Le deuxième principe - à mémoire banalisée - est envisagé avec un circuit de transfert de blocs de données DMA et un contrôleur de CRT type 8275 de chez Intel.

Tableau comparatif :

1er principe	2ème principe
. Temps d'occupation du bus pour le rafraîchissement d'écran ≈ 0 % . Possibilité d'adressage de 16 kØ octet de mémoire d'écran, soit 4 pages de données et d'attributs visuels associés. . Logique spécifique : - Multiplexeur (14 bits) - Génération des horloges de multiplexage - Adaptateurs de bus bidirectionnels pour les données (8 bits)	. Temps d'occupation du bus pour le rafraîchissement d'écran de 11 à 15 % suivant le type de DMA. . Possibilité d'adressage d'une mémoire d'écran égale à la mémoire disponible sur le bus d'adresse (peut être supérieure à 16 kØ). . Logique spécifique : - Un circuit type DMA - Un verrouillage d'adresses DMA (8 bits) - Un adaptateur de bus bidirectionnel pour les données DMA (8 bits) - Un multiplexeur de commandes lecture/écriture microprocesseur ou DMA.

Il apparait que la supériorité du point de vue performances du premier principe, est compensée par une limite à l'extension mémoire (cas des fonctionnements multi-pages). Au point de vue coût, les deux principes se valent, ce qui amène à prolonger la comparaison à un niveau d'étude plus élargi. Chaque principe est en effet lié à une logique de visualisation qui lui est propre.

3.3. Deux projets de réalisation de la logique de visualisation

3.3.1. Etude de la logique de visualisation avec mémoire d'écran spécialisée (figure 10)

Une logique de visualisation avec rafraîchissement d'image par mémoire spécialisée nécessite donc :

- L'utilisation d'un contrôleur de CRT (ou de la logique câblée équivalente mais plus couteuse) qui génére les adresses de rafraîchissement .
- Un système de multiplexage d'adresses entre le contrôleur de CRT et le microprocesseur.

. Rôle du contrôleur de visualisation

Le contrôleur 6845, adapté au type de rafraîchissement, assure les fonctions suivantes :

- . Dialogue avec le microprocesseur
- . Adressage de la mémoire d'écran
- . Comptage des lignes vidéo par rangées de caractères
- . Génération de signaux de visualisation.

- Le contrôleur, circuit en partie programmable, est relié au bus de données et de contrôles afin qu'à la mise sous tension de l'appareil, le microprocesseur vienne charger les paramètres qui caractérisent la visualisation d'écran :

  - . Nombre de caractères par rangée (NC)
  - . Nombre de lignes par rangée (NL)
  - . Nombre de rangées par écran (NR)

- Déclenchement et durée des signaux de retours de balayage
- Position du curseur sur l'écran.

- En cours de programme le microprocesseur peut lire des registres donnant la position du curseur ou d'un photostyle.

- Le contrôleur réalise à partir des paramètres "NR " et "NC " le comptage d'adresses mémoire pour le rafraîchissement.

- Il génère à partir du paramètre'NL' le comptage d'adresse ligne nécessaire au générateur de caractères.

- Il génère enfin des signaux agissant directement à l'entrée du moniteur vidéo : signal d'inhibition vidéo pour les retours de balayage du spot, signal de visualisation du curseur et signaux de déclenchement des retours de balayage, horizontal et vertical.

. Transfert des informations

- Adressage de la mémoire d'écran : l'adresse du caractère à visualiser est fournie par le contrôleur au travers d'un multiplexeur de deux fois n bits :
  de 11 bits pour une mémoire de 2 x 2 kø (1 page caractères + attributs),
  jusqu'à 14 bits pour une mémoire 2 x 8 kø (4 pages maximum).

  La mémoire est donc partagée entre la mémoire ' caractères ' (2000 mots par page) et la mémoire ' attributs ' (2000 mots par page), les deux mémoires sont adressées en parallèle (2x2 kø ⇒11 bits) puisqu'elles génèrent des données agissant simultanément. Une seule page de mémoire peut être sélectionnée à la fois.

  Pour une adresse donnée la mémoire ' caractères ' génère en sortie un mot ' caractère ' de 8 bits (256 différents) et la mémoire ' attributs ' un mot ' attribut ' de 8 bits également.

- Adressage de la mémoire générateur de caractères :

  les deux mots de la sortie mémoire d'écran sont verrouillés du côté vidéo afin qu'ils ne soient pas perturbés par d'éventuels accès à la mémoire d'écran par le microprocesseur.

  A la sortie des verrous le mot ' attribut ' est directement appliqué à la logique de mixage vidéo et le mot ' caractère ' fournit 8 bits d'adresse au générateur de caractères. Les 4 autres bits sont fournis par le comptage de ligne qui permet de déterminer le segment de la matrice de caractère (fig. 11).

L'adresse complète du générateur de caractères donne en sortie un code de 8 bits correspondant à l'affichage possible de 8 points sur un segment de ligne ; un bit à ' 1 ' peut représenter un point de phosphore excité sur l'écran.

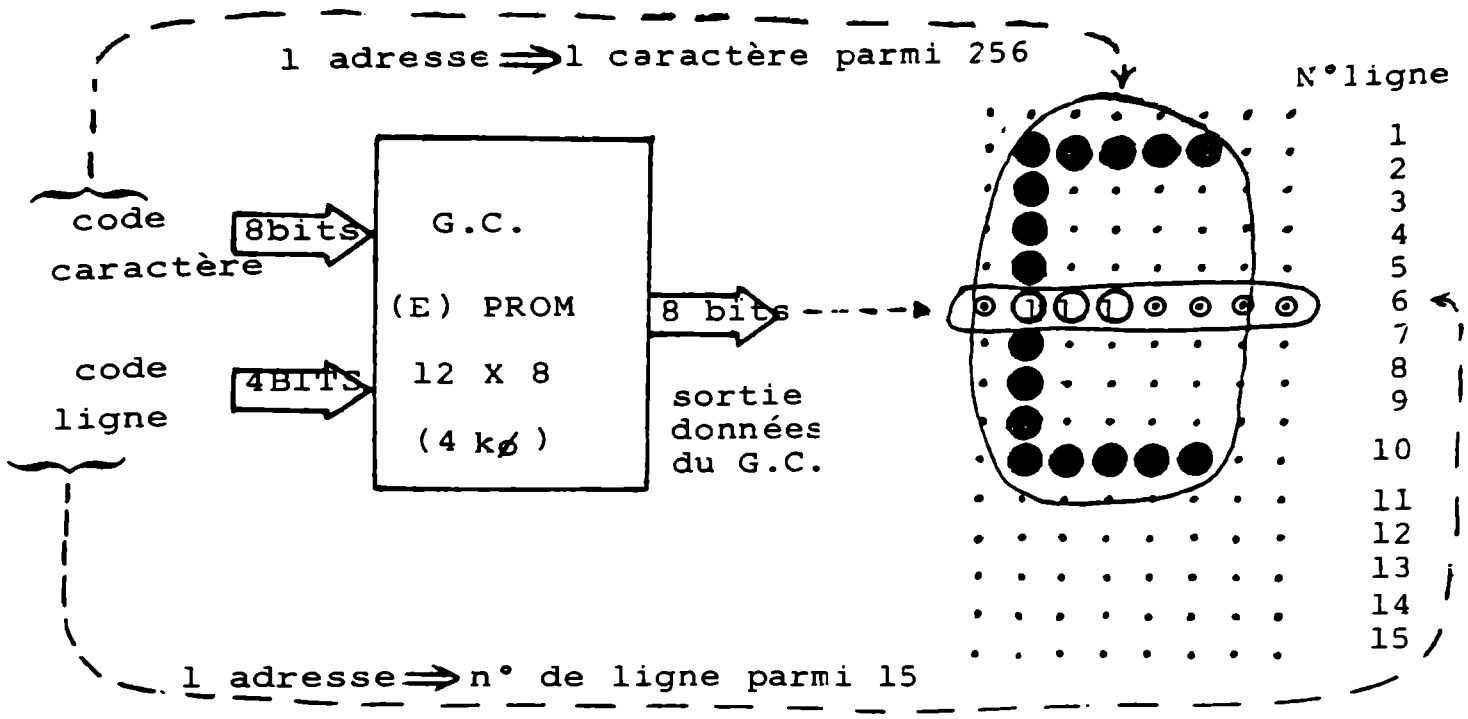


Figure 11 : ADRESSAGE DU GENERATEUR DE CARACTERES

- Sérialisation et mixage vidéo :

Le mot affichable peut être chargé en parallèle dans un registre à décalage. Le décalage série s'effectue au rythme d'une horloge ' point ', dont la période correspond à l'espace parcouru par le spot entre deux points d'excitation de l'écran.
La sortie du registre donne le ' signal vidéo '.

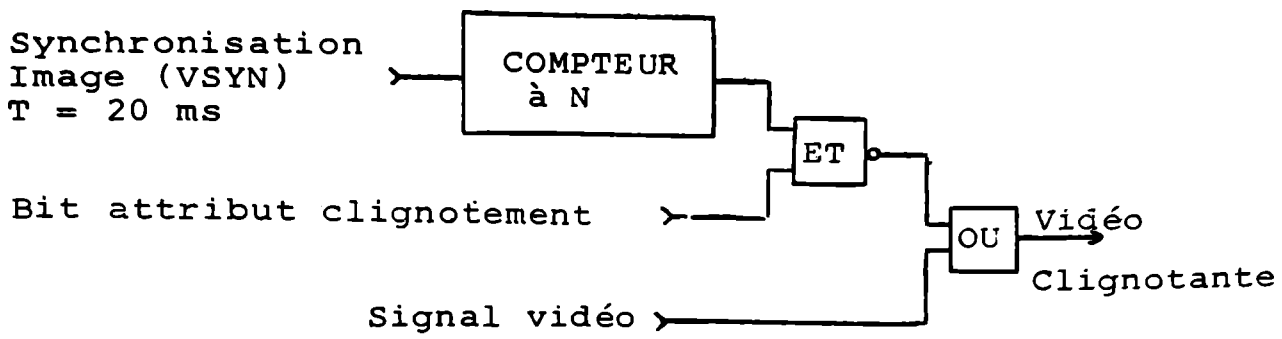
Plusieurs mixages sont réalisés sur ce signal avec des signaux d'attributs visuels et le signal curseur.

Les mixages possibles sont :

- Le forçage du signal à '0 ' ⟶ suppression vidéo
- Le forçage du signal à '1 ' ⟶ vidéo continue
- La variation du niveau de tension du signal vidéo.

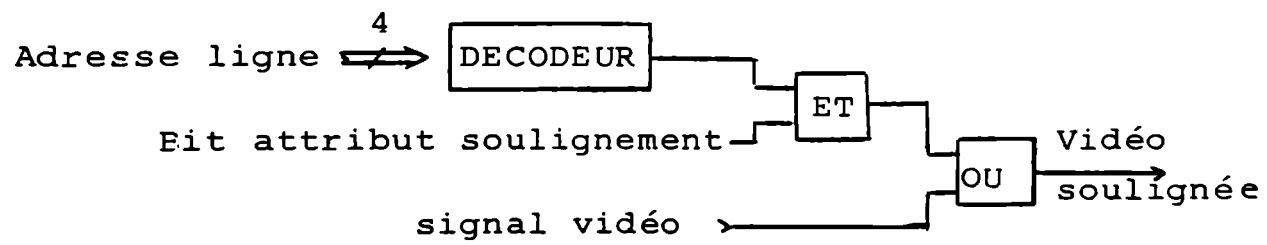
Les attributs visuels peuvent être réalisés de la façon suivante :

- Le clignotement :

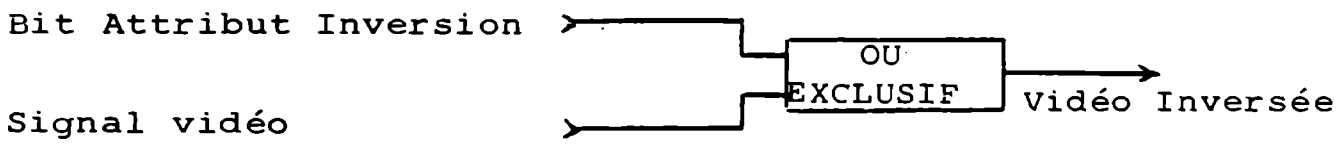


Exemple : N = 16 ⟹ 3 clignotements par seconde.

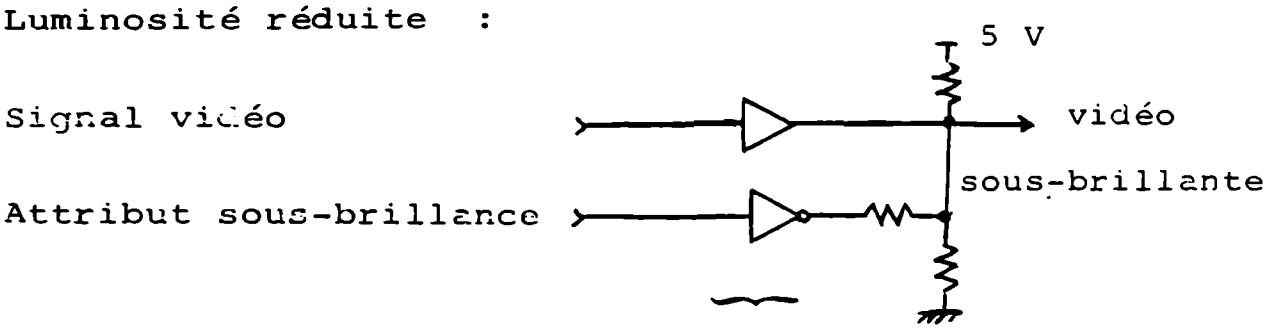
- Le soulignement :



- L'inversion vidéo :



- Luminosité réduite :



Portes de puissance

. <u>Séquencement des opérations</u> :

Le séquencement nécessite la réalisation d'un système d'horloges à partir de l'horloge de base, c'est à dire l'horloge ' point ', générée par un oscillateur.
Soit TP la période de l'horloge ' point ' et TC celle de l'horloge ' caractère ' (TC = nTP , n étant le nombre de points horizontaux de la matrice de caractère).

Les calculs d'horloges dépendent des caractéristiques de propagation dans les circuits utilisés; toutefois la logique doit respecter les 2 conditions absolues suivantes :

- Tous les accès à la mémoire écran doivent être exécutés dans un temps $\Delta t \leqslant \frac{TC}{2}$

- Chaque propagation des informations entre deux étages de mémorisation de la logique de rafraîchissement doit s'effectuer dans un temps $\Delta\tau \leqslant TC$.

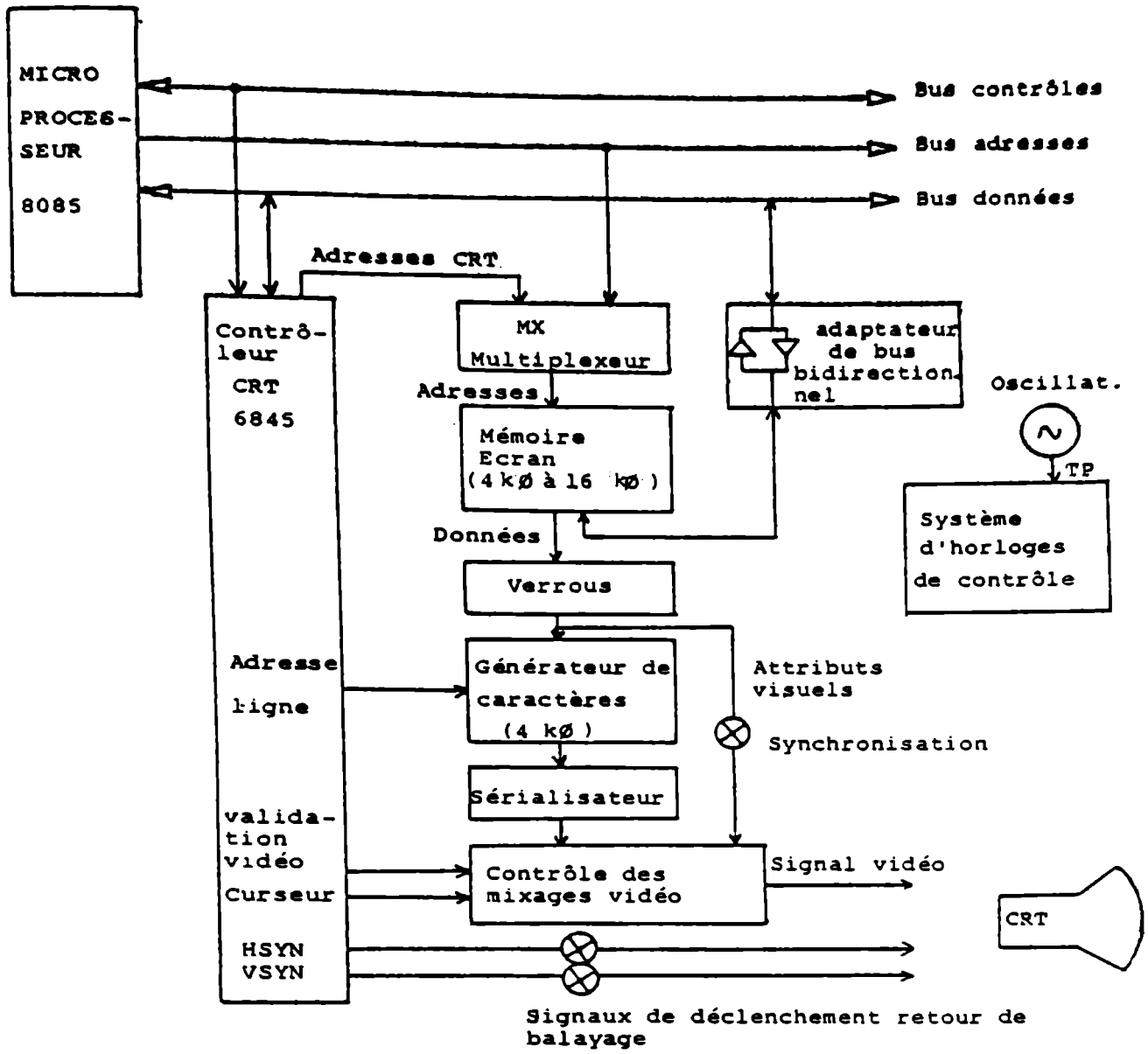


Figure 10 : SYNOPTIQUE DE LA LOGIQUE DE VISUALISATION

PROJET N° 1

3.3.2. Etude de la logique de visualisation avec mémoire d'écran banalisée : (figure 12)

Il a été vérifié que la meilleure solution utilisant la technique de rafraîchissement d'image avec la mémoire banalisée consiste à utiliser un circuit DMA et un contrôleur de CRT adapté au DMA (type 8275 de chez Intel).

Les lectures et écritures de données dans la mémoire d'écran se déroulent, à partir des cycles correspondants du microprocesseur, comme pour une autre mémoire du sytème.

Le rafraîchissement est effectué par le contrôleur de CRT et géré par le DMA.

- Rôle du contrôleur de CRT :

Le contrôleur a deux FIFO de 80 octets pour les caractères et deux FIFO de 16 octets pour les attributs. Un couple de registres (80 + 16) fournit les codes à visualiser pendant que l'autre couple est chargé par le DMA.

Lorsque la visualisation d'une rangée de caractères est terminée, le contrôleur de CRT alerte le DMA pour qu'il commence le transfert de la rangée suivante.

Le contrôleur fournit d'autre part les adresses du générateur de caractères et réalise les fonctions de curseur, d'inhibition de la vidéo pour les retours de balayage, de clignotement, de soulignement ainsi que les signaux permettant de générer des codes graphiques (utilisés pour visualiser des tableaux sur l'écran). Il crée enfin les signaux de balayage, horizontal et vertical.
A l'initialisation, le microprocesseur charge dans des registres du contrôleur les paramètres définissant le nombre de caractères par rangée, de lignes par rangée, de rangées par écran, les largeurs des retours de balayage, la position du soulignement, la fréquence du clignotement, et les caractèristiques du curseur.

- Transfert des informations :

Une fois les informations de visualisation acheminées dans le contrôleur, l'adressage du générateur de caractères, la sérialisation et les mixages vidéo sont réalisés de façon semblable à ceux du projet précédent.

- Séquencement des opérations

L'horloge ' caractère ' , tc, est générée à partir d'un oscillateur, tp, et d'un compteur à n (nombre de points par caractère). Elle fournit le rythme du contrôleur de CRT, du sérialisateur pour le chargement parallèle et des synchronisations des attributs visuels.

Enfin, chaque propagation des informations au travers des circuits entre deux étages de mémorisation de la logique de rafraîchissement doit s'effectuer dans un temps $\Delta\tau < tc$.

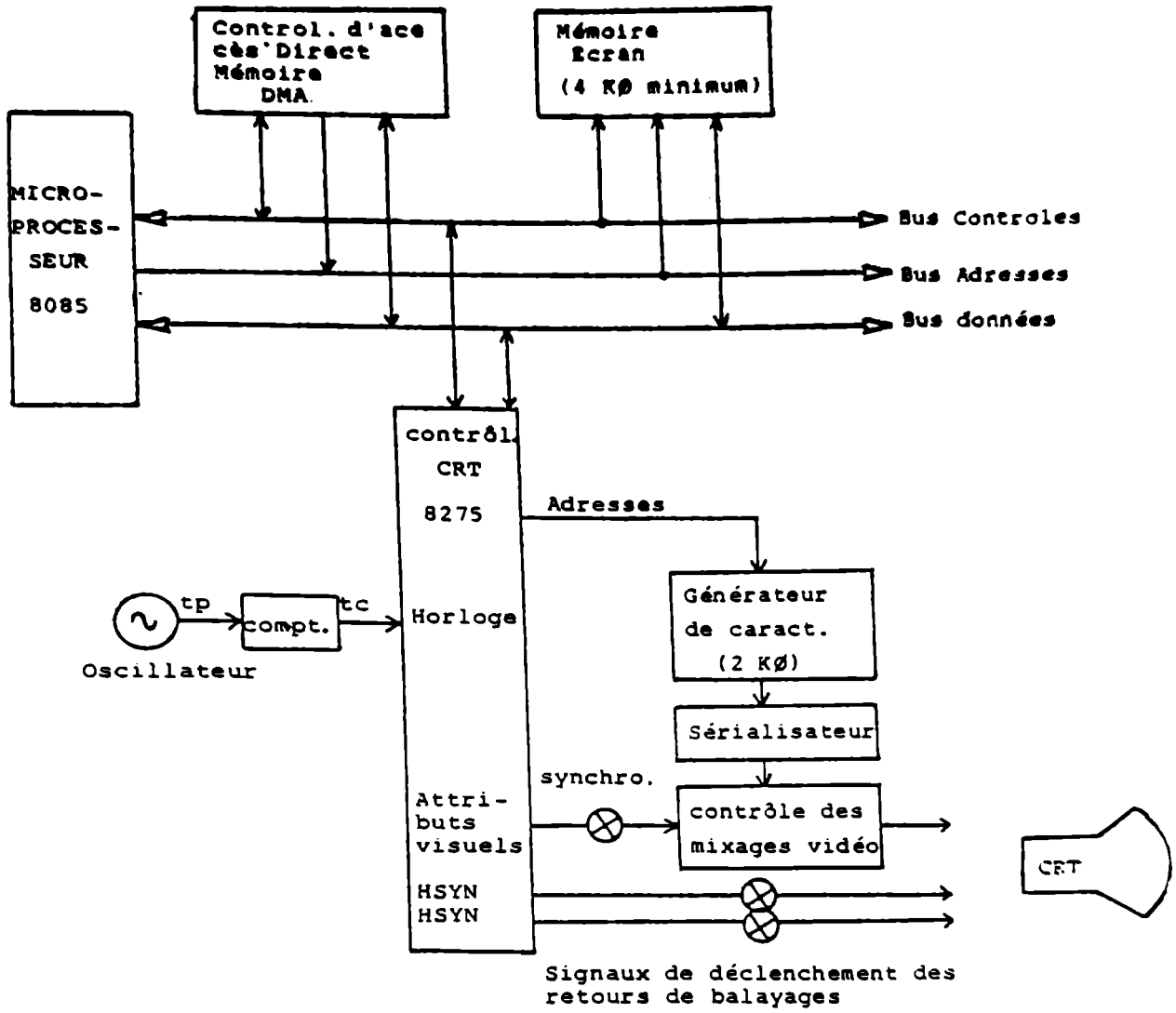


Figure 12 : SYNOPTIQUE DE LA LOGIQUE DE VISUALISATION

PROJET N° 2

3.4. Comparaison et choix

La comparaison des deux projets est traitée sous trois aspects :

- les performances
- le coût en matériel
- les fonctions remplies

3.4.1. Performances

Les performances du fonctionnement de la logique de visualisation s'évaluent principalement par le temps d'occupation (τ) du bus pour accomplir le rafraîchissement de l'image.

PROJET N° 1	PROJET N° 2
Temps d'occupation du bus pour le rafraîchissement $\tau \approx 0$ %	Temps d'occupation du bus pour le rafraîchissement : $\frac{(N1 + N2)\ N3}{Ti} \times t$ (cycle) N1 : nombre de cycles microprocesseur pour transférer 96 (80 + 16) octets (96 x 3 = 288 cycles avec le DMA 8237) N2 : nombre de cycles nécessaires pour initialiser un transfert DMA = 4 N3 : nombre de rangées de caractères 25 Ti : période image $= 20.10^{-3}$ d' où : $\tau = \frac{(288 + 4)\ 25}{20.\ 10^{-3}} 325.10^{-3} =$ 0,1186 $\tau \approx 12$ %

Il faut noter encore que l'utilisation d'un DMA dans le projet n° 2 permet de disposer d'autres canaux pour gérer d'autres transferts de blocs de données.

3.4.2. Coût en matériel

La comparaison des deux projets en nombre de circuits de type LSI et MSI met en évidence la différence de coût entre les solutions :

FONCTIONS	PROJET N° 1		PROJET N ° 2	
	CIRCUITS	Nbre	CIRCUITS	Nbre
Mémoire d'écran 4 kø	RAM rapide de 2 kø $t_{Acc} \leqslant \frac{Ic}{2} - tp\ (Mx)$ = 200 ns	2*	RAM de 2 kø	2 *
Chargement et lecture de données en mémoire	Multiplexeurs 8 bits Registres à bascules D, 8 bits, sorties 3états adapt. de bus 8 bits	3 2 2	DMA Regist.à bascule D, 8 bits,sort.3 états adapt.de bus 8 bits bidirectionnel	1 * 1 1
Contrôleur CRT	6845 MOTOROLA	1 *	8275 INTEL	1 *
Verrouillage	Registres 8 Bits	2		
Générat. de caract. et sérialisateur	PROM 2 kø Regist.à décal. 4 bits	2 * 3	(Idem	2 * 3
Mixage Vidéo	Décodeur 8 x 1 bits Synchro.Registre 8 bits Multiplex. 2 (4 x 1) Compteur à 16 16 portes logiques	1 1 1 1 4	(Idem 8 portes logiques	1 1 2
Horloges	12 portes logiques Oscillateur Regis.à décal. 4 bits 2 bascules type D	3 1 1 2	4 portes logiques Oscillateur Compteur à 16	1 1 1

TOTAL DES CIRCUITS	TYPE LSI *	5	6
	TYPE MSI	27	12

Sachant qu'un circuit LSI coûte 5 à 10 fois plus cher (montage compris) qu'un circuit MSI, la comparaison fait apparaitre un prix semblable pour les deux solutions. Le calcul détaillé aboutit à une différence de 9 % en faveur du projet n° 2.

3.4.3. Fonctions remplies

Réalisation des fonctions du terminal :

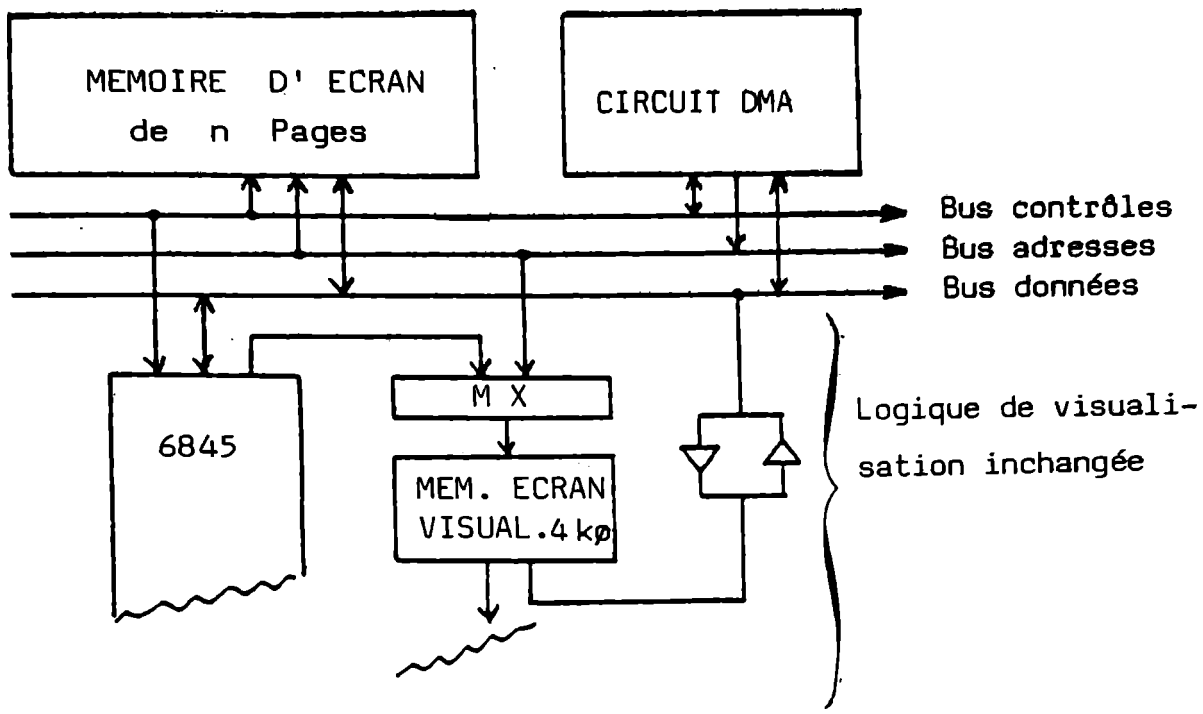
FONCTIONS DEMANDEES	FONCTIONS REALISEES	
	Projet N°1	Projet N° 2
Gestion d'un écran 15 pouces	OUI	OUI
Dimensions de la matrice de caractères	OUI	OUI
Générateur de 256 caractères différents	OUI	NON
Plusieurs visualisations du curseur	OUI	OUI
Types d'attributs de visualisation	OUI	OUI
Nombre d'attributs de visual. non limité	OUI	NON
Niveaux de luminosité et du son	OUI	OUI
Fonctions d'interfaces	OUI	OUI
Programme chargeable (émulation VU 3)	OUI	OUI
Fonctions classiques	OUI	OUI
Fonctions classiques	NON	OUI

Toutes les fonctions ne sont pas assurées par l'un ou l'autre des 2 projets, il est donc nécessaire d'étudier les solutions permettant de combler ces lacunes avant que de conclure sur le choix du projet.

- Solutions de réalisation de la fonction de ' partitionnement d'écran ' pour le projet n° 1 :

La fonction nouvelle à réaliser est le ' partitionnement d'écran' car, d'une part le contrôleur 6845 ne génère pas de signal d'interruption de fin de rangée de caractères nécessaire lorsque les zones mémoires à visualiser sont disjointes, et d'autre part la reprogrammation du contrôleur en cours de fonctionnement n'est pas garantie par le fabricant. Ceci est nécessaire pour modifier les registres d'adresse mémoire lorsque le rafraîchissement d'image passe d'une zone mémoire à une autre zone mémoire disjointe (cf. chapitre B.3.2.). Il faut noter également que le contrôleur 6845 qui a 14 bits d'adresse limite la mémoire d'écran à 4 pages (14 bits ⟶ 16 kø = 4 x 4 kø).

- Une 1ère proposition pour résoudre le problème du rafraîchissement en mode partitionnement d' écran consiste à ajouter à la logique de commande un circuit DMA qui, indépendamment du rafraîchissement, récompose, à partir des directives du microprocesseur, une mémoire d'écran à visualiser dont les adresses sont contigües.



NOUVEAU SYNOPTIQUE (Projet 1)

Le microprogramme gère une table qui permet au DMA de mettre à jour la mémoire d'écran visualisée à partir de la mémoire d'écran totale.

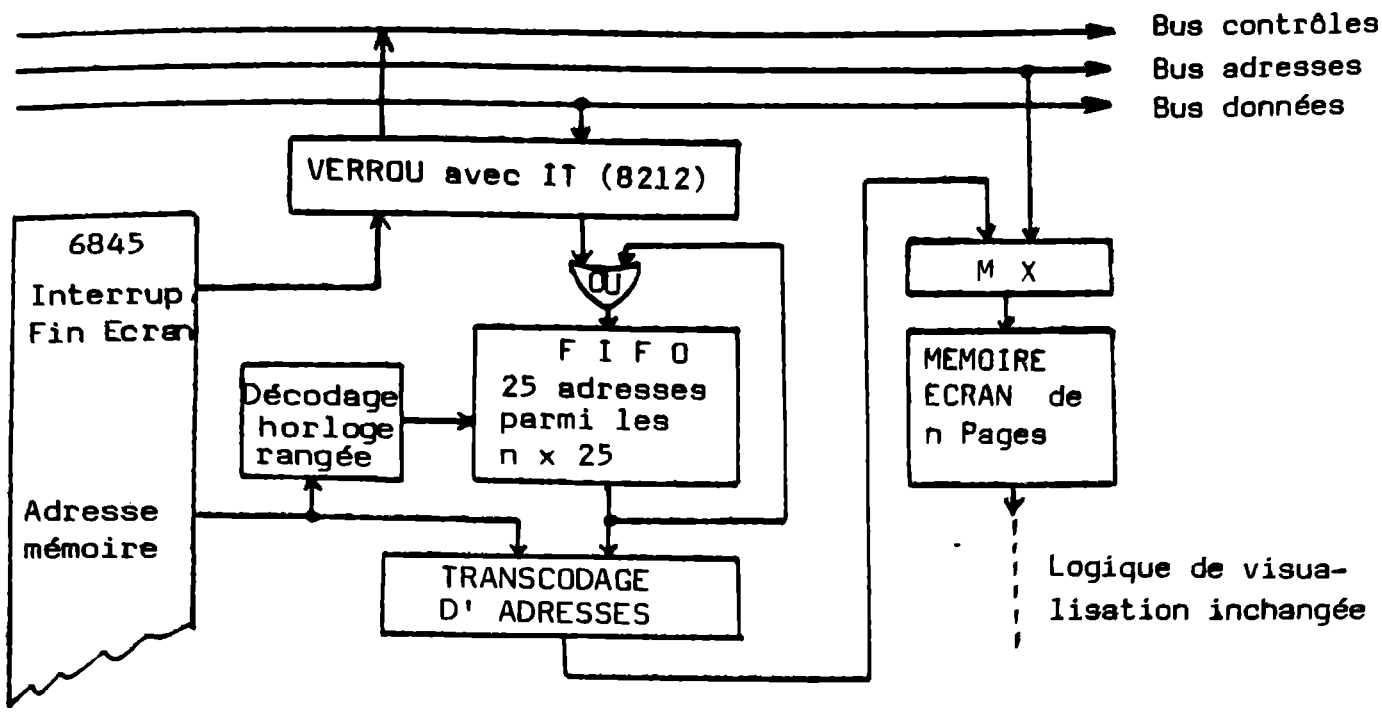
Le coût supplémentaire de cette première proposition est :

2 x 2 kø	de la mémoire écran à visualiser
1	circuit DMA
1	registre à bascules, 8 bits, trois états
1	adaptateur de bus, 8 bits, bidirectionnel

Le temps d'occupation du bus pour le rafraîchissement reste négligeable.

- Une seconde proposition consiste à combiner l'adresse fournie par le contrôleur 6845 avec une adresse de numéro de rangée de caractères parmi les n x 25 de la mémoire d'écran de n pages.

  A la fin du rafraîchissement d'image, pendant le retour de balayage vertical, le contrôleur émet une interruption qui autorise le microprocesseur, s'il y a lieu, à changer la configuration du partitionnement en chargeant de nouvelles adresses de rangées dans une mémoire FIFO.



NOUVEAU SYNOPTIQUE (Projet 1)

Pendant le rafraîchissement la mémoire FIFO est bouclée sur elle-même au rythme du balayage ' rangée de caractères ' et fournit à chaque période une adresse rangée qui, transcodée avec l'adresse du contrôleur, délivre l'adresse finale du mot à visualiser.

Le coût supplémentaire de cette deuxième proposition est :

1	verrou avec IT, 8 bits (type 8212 de chez Intel)
2	multiplexeurs 4 fois - 2 x 1 bits
1	FIFO de 25 octets minimum
1	décodeur 8 x 1 pour générer l'horloge ' rangée '
3	PROMs pour le transcodage : (2 K x 8) + 2 (1 K x 4)
1	boitier de portes logiques.

L'occupation du bus pour le rafraîchissement reste négligeable

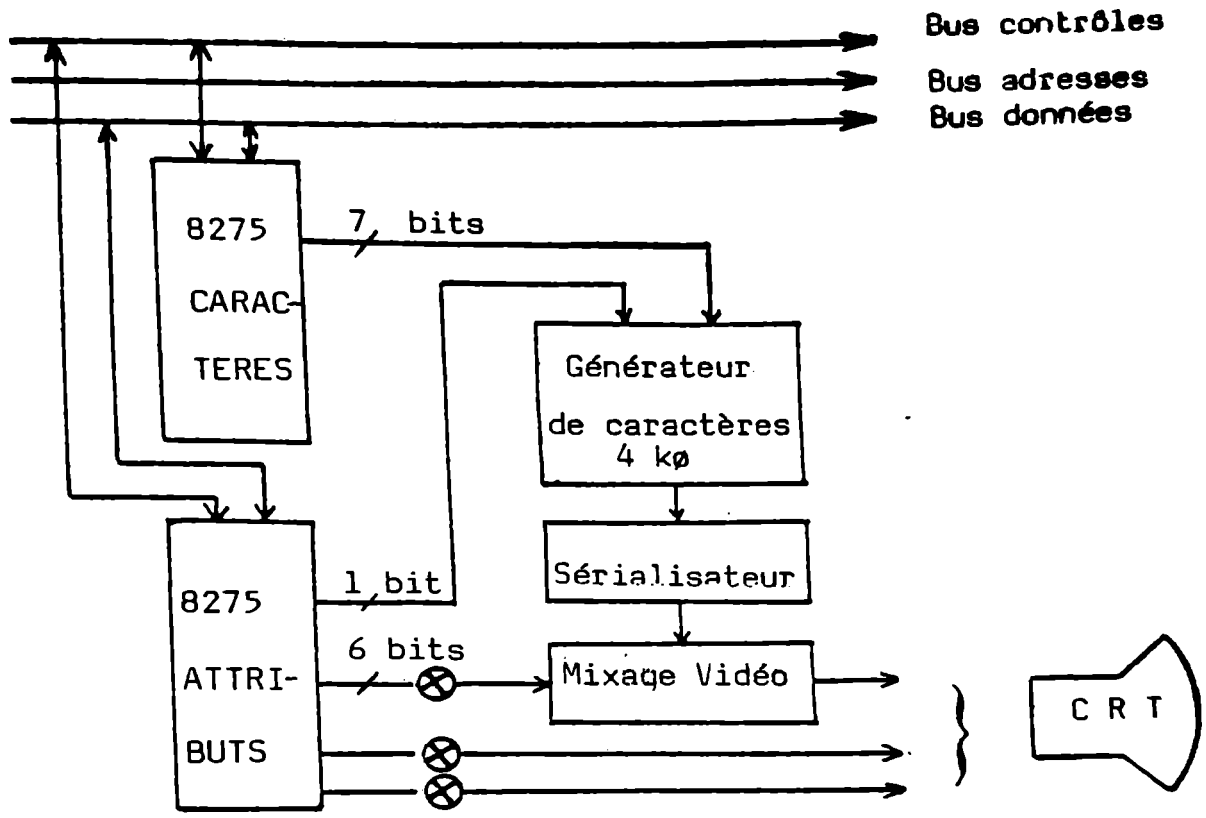
- Solution pour réaliser les 80 attributs visuels par rangées de caractères et le générateur de 256 caractères dans le projet n° 2.

La conception même du contrôleur 8275 limite le nombre d'attributs visuels à 16 par rangée de caractères (1 FIFO de 80 octets pour les caractères et 1 FIFO de 16 octets pour les attributs). Il est donc impossible d'associer à chaque caractère de l'écran un attribut de visualisation ; et par là même il est impossible d'étendre le générateur de caractères de 128 à 256 graphismes différents. En effet, ceci n'est possible qu'en utilisant un des bits du code attribut pour l'affecter au codage du caractère ce qui donne (7 + 1) bits soit 256 codes de caractères différents.

La solution la moins onéreuse pour résoudre ces deux problèmes consiste à utiliser deux contrôleurs de CRT 8275 en parallèle afin de disposer de deux FIFO de 80 octets. Le premier contrôleur gère les caractères et le second les attributs visuels plus un bit pour les caractères.

Cette solution n'entraine en fait que le coût supplémentaire d'un circuit LSI. D'autre part, il n'existe pas actuellement d'autres contrôleurs de ce type permettant de stocker, sous contrôle DMA, 160 octets par rangée de caractères.

Il faut remarquer enfin que le deuxième contrôleur permet d'améliorer le système et de l'ouvrir vers de nouvelles fonctions notamment pour la programmation.



NOUVEAU SYNOPTIQUE (Projet 2)

Le coût supplémentaire de cette solution est :

1 contrôleur de CRT ' 8275 '

Les performances sont modifiées par :

Le temps d'occupation du bus pour effectuer le rafraîchissement qui devient :

$$\frac{(160 \times 3 + 4)\ 25}{20.10^{-3}} \times 325.10^{-9} = 0{,}1966 \text{ , soit :}$$

20 % au lieu de 12 %

L' étude comparative des deux projets en nombre de circuits et coût devient donc :

DESIGNATION	PROJET N° 1		PROJET N° 2
	AVEC DMA	AVEC FIFO	
Nombre de L S I	5 + 3 = 8	5 + 4 = 9	6 + 1 = 7
Nombre de MSI	27 + 2 = 29	27 + 5 = 32	12 + 0 = 12
Coût Réel Indiciel 1981	143	132	100
Charge Bus pour le rafraîchissement	0 %	0 %	20 %

CONCLUSIONS :

Sans la réalisation des fonctionnements en ' multi-page ' et en ' partionnement ' le projet d'étude avec mémoire d'écran spécialisée est très intéressant pour de meilleures performances et un prix équivalent au second projet. Dans le cadre des objectifs à réaliser il est demandé que les deux fonctions citées soient assurées par le matériel. La comparaison des deux projets réalisant toutes les fonctions fait retenir, pour des raisons de coût et d'encombrement du circuit imprimé, la solution avec mémoire d'écran banalisée.

C'est donc cette solution qui est développée par la suite.

Parmi les autres choix présentant un intérêt dans le contexte de l'étude deux seront développés :

- le choix des mémoires
- le choix du circuit d'accès mémoire direct (DMA)

4. CHOIX DES MEMOIRES

Les mémoires utilisées doivent satisfaire à des contraintes de fonctionnement (lecture, lecture et écriture, compatibilité microprocesseur), de capacité, de performance (temps de traversée) et de caractéristiques électriques (alimentation, consommation, charges capacitives, ...).

Lorsque plusieurs types de circuits satisfont à ces contraintes, les moins coûteux , les plus fiables et les plus disponibles sur le marché sont retenus en priorité.

Le tableau de la figure 13 présente les types de mémoires actuellement disponibles sur le marché.

4.1. La mémoire de commande

4.1.1. Mémoire morte (VU1, VU2 et VU4)

Cette mémoire qui stocke les codes de commandes du terminal et qui constitue le microprogramme, est généralement de type ' mémoire morte '. Le microprogramme ou ' firmware ' est, en effet, figé et fait partie intégrante de l'appareil.

Les caractéristiques de cette mémoire sont :

- lecture uniquement ⟶ type ROM, PROM ou EPROM
- capacité maximum de 18 kø
- connexion possible aux bus du microprocesseur
- temps de traversée imposés par le µprocesseur 8085 A :

  tp1* : (sorties des données / adresses 8085) < 500 ns

  tp2* : (sorties des données / signal de lecture 8085) < 250 ns

(*) tp = temps de propagation des mémoires et de la logique d'accès
signaux d'adresse du 8085 ⇒ signal de sélection de la mémoire
signal de lecture du 8085 ⇒ signal de validation de la mémoire

- Les mémoires mortes sont de type MOS car les mémoires bipolaires ont une consommation en courant trop élevée pour les bus d'adresses et de données.

  L'emploi de ROMS MOS peut être envisagé pour des quantités supérieures à 10.000 mémoires identiques.

  A partir de ces quantités le coût du masque de fabrication est amorti. Ce choix interdit toutefois une évolution éventuelle du contenu des ROMS (programme figé).

  Le type de mémoire retenu est en fait l' EPROM dont le coût par bit diminue plus vite que celui des ROMS MOS. L'absence d'investissement dans la réalisation d'un masque autorise une éventuelle évolution des microprogrammes en cours de fabrication du terminal.

- Le découpage de l'espace mémoire du système réserve une capacité maximale de 18 kø à la mémoire de commande. La capacité utilisée varie suivant le type d'appareil à émuler. Pour cette raison les EPROMS choisies sont de 2 kilo-adresses (11 bits) et de 8 bits de données en sortie. Un microprogramme de 10 kø est ainsi contenu dans 5 boitiers de mémoire.

- La mémoire choisie est la mémoire 2716 (2 K x 8) de chez Intel dont les temps de traversées sont :

  tp1 : (données/ signal de sélection) < 450 ns
  tp2 : (données/ Signal de validation) < 120 ns

4.1.2. Mémoire vive

Pour réaliser le fonctionnement du terminal VU 3, le microprogramme doit être chargé dans la mémoire à partir d'un contrôleur de terminaux. Dans ce cas particulier la mémoire de commande est vive. Ses caractéristiques sont :

- écriture et lecture $\Longrightarrow$ type RAM
- capacité de 16 kø
- connexion possible aux bus du microprocesseur
- temps de traversée imposés par le microprogramme 8085 A

  tp1 < 500 ns ; tp2 < 250 ns
  tp3 (durée du cycle d'écriture) < 650 ns (cf. annexes)
  tp4 mini (front actif du signal d'écriture / données)
  < 390 ns.

Les mémoires de la figure 13 qui répondent à ces caractéristiques sont la RAM MOS statique de 8 ou 16 kilo-bits et la mémoire RAM MOS dynamique de 16 kilo-bits.

Les RAMS dynamiques n'ont qu'un bit de sortie pour 16 kilo-adresses, alors que les RAMS statiques possèdent 8 bits de sortie pour 2 kilo-adresses ; mais pour réaliser 16 kø de mémoire le nombre de boitiers nécessaires est le même.

En fin de compte, le critère de choix entre les deux types de mémoire est le prix. Une étude du coût en fonction de la capacité (figure 14) montre que le choix de la mémoire dynamique est plus avantageux pour une capacité supérieure à 8 kilo-octets.

C'est le choix de ce type de mémoire, dont la structure ne présente qu'un bit en sortie et dont l'utilisation nécessite une logique de rafraîchissement, qui amène à développer un circuit imprimé légèrement différent pour l'émulation du terminal VU 3.

Les mémoires retenues sont les RAMS 4116-2 (16 K x 1) de chez Mostek dont les temps de traversées sont :

tp1 (données / signal de sélection) < 150 ns + couches logiques
tp3 (durée du cycle d'écriture) = 320 ns + couches logiques
tp4 Mini (front d'écriture / données) = 45 ns

4.2. Mémoire du générateur de caractères

Cette mémoire contient les codes des caractères affichables qui sont directement transmis sur le signal vidéo.

Ses caractéristiques sont :

- mémoire morte PROM ou EPROM.
- capacité :

  8 bits pour adresser un caractère parmi 256 possibles et 4 bits pour adresser le rang de la ligne de balayage, soit 12 bits d'adresse pour 8 bits de données en sortie (4 kø)

- temps de traversée :

  Ce temps d'accès doit être inférieur à la période de l'horloge caractère moins le temps de traversée de couches logiques soit :

  Tp < Tc = 458 ns - 50 ns = 408 ns

Le choix se porte sur des mémoires EPROM pour les raisons déjà évoquées. Le type retenu est l' EPROM 2716-2 (mémoire rapide) de chez Intel dont le temps d'accès tp est tel que : tp < 390 ns.

4.3. Mémoire de travail et mémoire d'écran

Ces mémoires, qui contiennent des données variables, ont les caractéristiques suivantes :

- écriture et lecture ➡ type RAM
- capacité de 2 kø suffisante pour la mémoire de travail et de 4 kø pour la mémoire d'écran (2000 caractères + 2000 attributs visuels) ; dans le cas particulier d'un microprogramme avec fonction 'multi-pages' la capacité d'écran s'étend à n x 4 kø pour n pages.

- connexion possible aux bus du microprocesseur
- temps de traversée imposés par le microprocesseur 8085 A et par le DMA 8257-5 (voir annexe).

tp1	<	500 ns	pour le 8085 A
tp2	<	250 ns	
tp3	<	650 ns	
tp4 mini	<	390 ns	

tp1	<	200 ns	pour le 8257-5
tp2	<	200 ns	
tp3	<	650 ns	
tp4 mini	<	350 ns	

- Dans le cas général la mémoire choisie est la RAM MOS statique de 2 kø par boitier (3 boitiers), en l'occurence la 4802 de chez Toshiba dont les temps de traversées sont :

tp1	<	200 ns
tp2	<	120 ns
tp3	<	200 ns
tp4 mini	<	60 ns

- Pour des terminaux réalisant la fonction ' multi-pages ', il est préférable d'utiliser des mémoires RAMS dynamiques puisque dans ce cas la capacité totale de RAMS dépasse 8 kø (figure 14.)

4.4. Mémoire de sauvegarde

Cette mémoire contient, entre autres, les informations donnant la configuration du terminal. Son alimentation commute sur une batterie à la détection d'une baisse de tension. Ses caractéristiques sont :

- mémoire à écriture et à lecture (données chargeables par un opérateur via le clavier et le microprocesseur).
- une capacité de 512 octets permet de mémoriser les mots de configuration, de réglages de luminosité et d'alarme sonore, et éventuellement un type de générateur de caractères chargeable.
- connexion possible aux bus du microprocesseur
- temps d'accès imposés par le microprocesseur 8085 A :

tp1	<	500 ns
tp3	<	650 ns
tp4 mini	<	400 ns

- consommation de repos adaptée aux débits des accumulateurs

. La mémoire retenue est donc de type RAM CMOS statique, de capacité de 1 kilo-adresses pour 4 bits de données (4 Kbits), et plus précisément du type TC 514 P de chez Toshiba. Cette mémoire a des temps d'accès :

tp1	<	450 ns
tp3	=	450 ns
tp4 mini	=	200 ns

et sa consommation au repos est de 60 µA maximum, et 6 µA typique.

. Il faut noter enfin que l'apparition prochaine de modèles de mémoire morte ' EAROM ', électriquement effaçable et reprogrammable, associée dans un même circuit à une RAM permettra de remplacer le dispositif de sauvegarde actuel : RAM + batterie.

Famille	Type	Capacité en kilo-bits	Temps d ' accès	Remarques
Mémoires Mortes (lecture)	ROMS Bipolaires			Suplentées par les PROMS bipolaires à fusibles
	PROMS Bipolaires à fusibles	16 K (TTL) 1 K (ECL)	35 à 100 ns 25 ns (ECL)	ECL ⟹ Tensions particulières
	ROMS MOS	64 K	500 ns	Coût du masque de fabrication à amortir.
	EPROMS	32 K 16 K	1 µs 500 ns	Modification du contenu par U.V.
Mémoires Vives (lecture et écriture)	RAMS STATIQUES MOS	4 K 8 et 16 K	25 ns 200 ns	
	RAMS STATIQUES C-MOS	4 et 8 K	500 ns	Faible consommation : 8 K { 100 µw au repos 180 mw active
	RAMS STATIQUES Bipolaires	1 et 4 K	30 à 50 ns (TTL) 20 ns (ECL)	ECL ⟹ Tensions particulières
	RAMS Dynamiques MOS	16 K	150 à 200 ns	. 3 tensions : + 5V, + 12V et - 5 V . Cycles de rafraichissement . Sortie 1 bit
		64 K		Début de commercialisation

Figure 13 : CARACTERISTIQUES DES MEMOIRES STANDARDISEES DE GRANDES CAPACITES

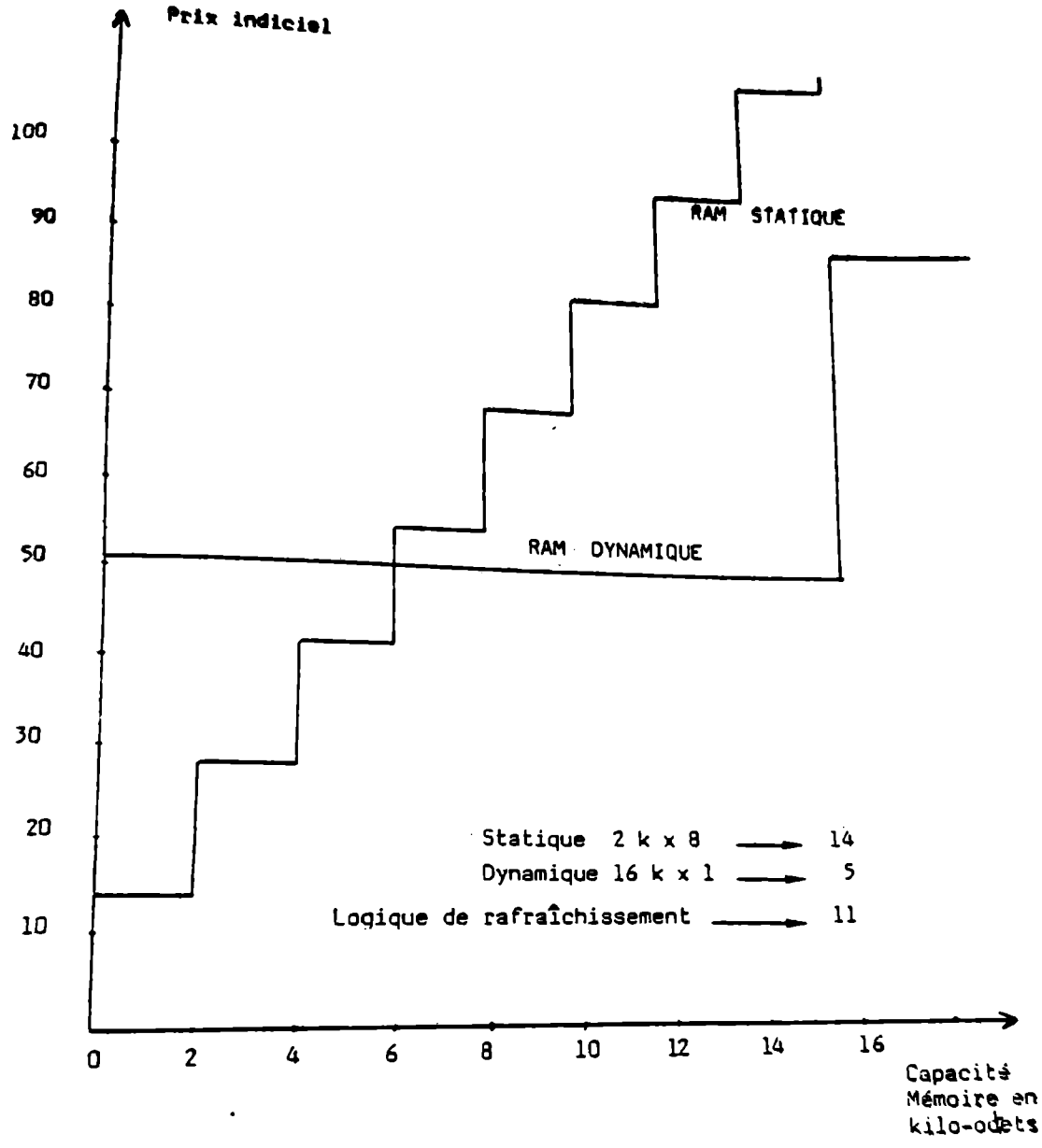


Figure 14 : COUTS DES MEMOIRES RAMS MOS ' STATIQUES ' et ' DYNAMIQUES '.

(Estimation 1982)

5. CHOIX DU CONTROLEUR D' ACCES DIRECT MEMOIRE (DMA)

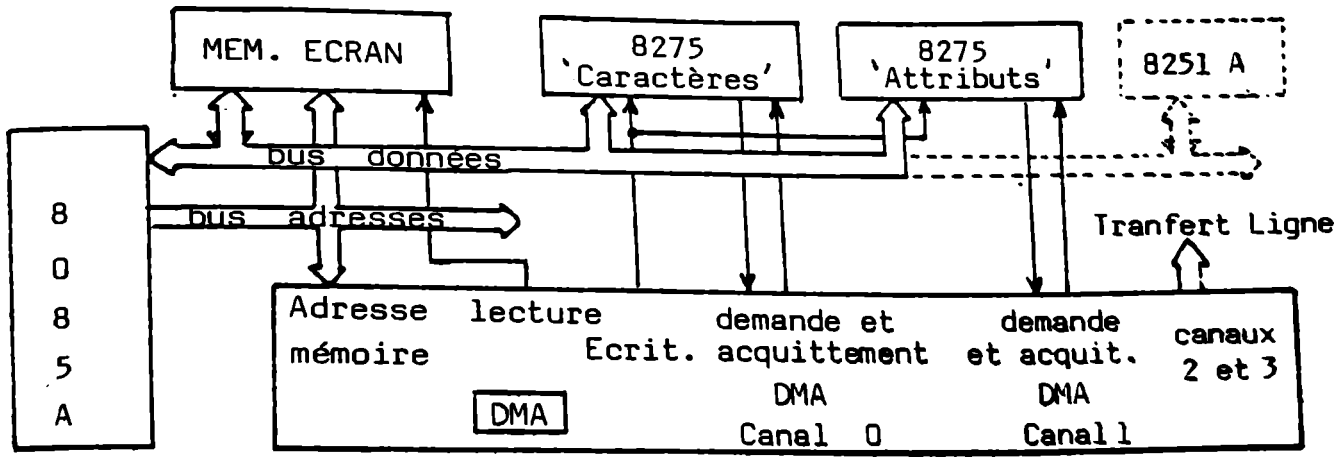
5.1. Rappel

Ce circuit a été incorporé dans la logique de commande pour réaliser le transfert des données émises ou reçues sur la ligne de transmission et le rafraîchissement de l'écran.

Sa fonction principale est d'améliorer les performances. Le premier type de transfert n'est utilisé efficacement qu'en mode de transmission par blocs de données et représente un taux d'occupation du bus très faible devant celui du rafraîchissement d'écran qui est de 20 % avec le circuit DMA le plus rapide.

C'est donc en tenant compte principalement de la fonction de rafraîchissement que sera choisi le type de DMA.

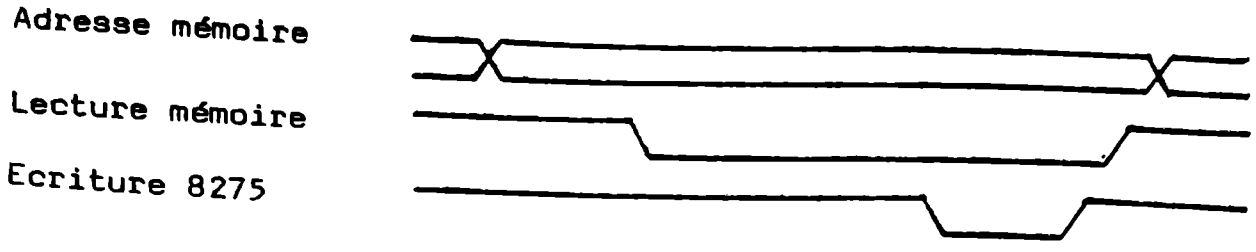
5.2. Principe de fonctionnement du DMA



Le microprocesseur initialise le DMA pour chaque transfert de bloc de données avec un sens de transfert, une première adresse de mémoire et un nombre d'octets à transférer entre une mémoire et un périphérique.

Le DMA se substitue au microprocesseur pour effectuer un transfert pour le rafraîchissement d'écran lorsque le contrôleur 8275, conçu pour ce type de fonctionnement, en fait la demande.

Les signaux de contrôle d'un transfert sont :



Lorsque le transfert est terminé le DMA restitue le contrôle du bus au microprocesseur.

Le transfert d'un octet s'effectue ainsi dans un temps de 3 ou 4 périodes d'horloge suivant le type de DMA.

5.3. Deux types de D M A .

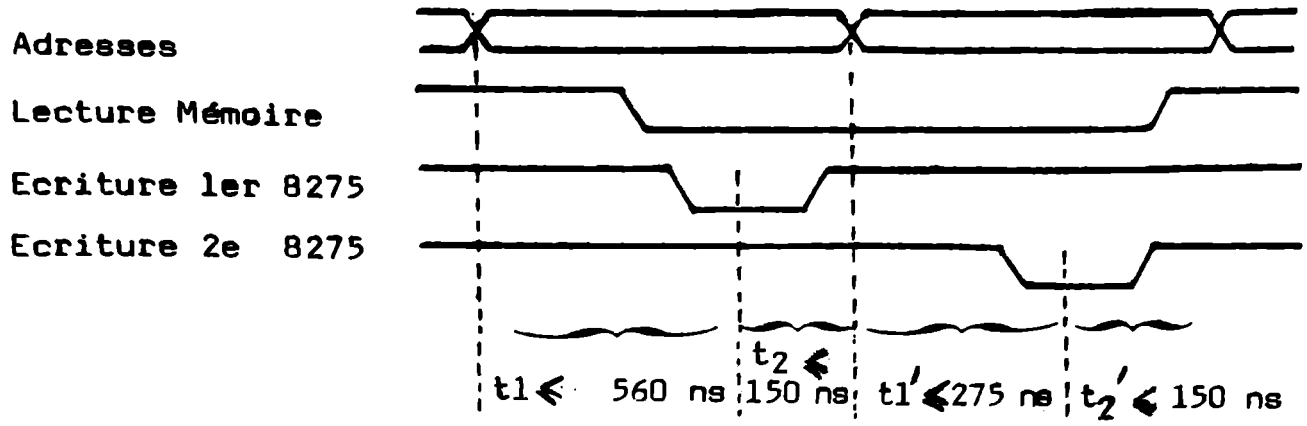
Le choix du contrôleur DMA se porte implicitement sur le fournisseur Intel afin d'assurer la compatibilité avec le contrôleur de TRC (8275) sans adjonction de logique ou de programme. Deux types de DMA sont offerts dont la différence, en ce qui concerne l'étude, réside dans la durée des cycles de transferts :

- le premier circuit, 8257, réalise le transfert d'un octet en 4 périodes d'horloge (1,3 µs)
- le second circuit, 8237, le réalise en 3 périodes d'horloge (0,975 µs).

A priori le 8237 est plus intéressant et permet un taux d'occupation du bus de 20 % pour le rafraîchissement. Toutefois, ce taux élevé peut être sensiblement amélioré en réalisant une double écriture des 8275 pendant un même cycle de transfert. Ceci est assez facilement réalisé par de la logique supplémentaire en forçant un multiplexage d'adresse pour passer de la mémoire ' caractères ' à la mémoire ' attributs ', car le transfert constitue une alternance entre les octets de ' caractères ' et les octets d'' attributs ' associés. Parallèlement, deux impulsions d'écriture sont générées, une pour chaque contrôleur 8275.

Pendant un cycle de DMA, un caractère et l'attribut associé sont transférés et le nombre de transferts du DMA, pour rafraîchir une rangée de caractères, passe de 160 à 80.

Nouvelle forme des signaux de contrôle :



t1,t1' : accès mémoire + couches logiques

t2, t2' : écriture / stabilité des données

La durée maximale d'un tel cycle est de 1, 135 µs. Le temps de transfert peut dépasser la micro-seconde et seul le contrôleur DMA de type 8257 peut assurer ce fonctionnement.

5.4. Performances et choix

Le taux d'occupation du bus pour le rafraîchissement d'image est :

$$\tau = \frac{\left[(N1 \times N2 + N3)\ n\right]\ TcLK}{Ti}$$

avec :

N1	:	nombre de transferts par rangée de caractères
N2	:	nombre de rangées de caractères par écran : 25
N3	:	nombre de cycles d'initialisation DMA : 4
n	:	nombre de périodes d'horloge
TcLK	:	période de l'horloge : 0, 325 µs
Ti	:	période d'image : 20 ms

./..

<u>1ère solution</u> : Utilisation du DMA type 8237 avec une écriture par cycle de transfert :

$$\left.\begin{array}{l} n = 3 \\ N1 = 160 \end{array}\right\} \Longrightarrow \quad \tau = \frac{(160.25 + 4)\ 3 \times 0{,}325}{20.10^3} = 0{,}195$$

$$\boxed{\tau \approx 20 \ \%}$$

<u>2ème solution</u> : Utilisation du DMA type 8257 et d'une logique supplémentaire (*) (5 boitiers TTL)

$$\left.\begin{array}{l} N1 = 80 \\ n = 4 \end{array}\right\} \Longrightarrow \tau = \frac{(80.25 + 4)4 \times 0{,}325}{20.10^3} = 0{,}13$$

$$\boxed{\tau = 13 \ \%}$$

Des deux solutions, c'est la seconde qui est retenue car, pour un faible coût en matériel supplémentaire, elle descend de 7 % le taux d'occupation du bus pour le rafraîchissement d'image. Cette amélioration des performances est utile, sinon nécessaire, pour garantir le fonctionnement du terminal aux vitesses de transmissions les plus élevées (> 20.000 bps).

(*) Les schémas et les chronogrammes sont donnés en annexe.

LOGIQUE DE COMMANDE :

FONCTIONNEMENT (HARDWARE)

Le synoptique (figure 15) donne l'organisation générale de la logique de commande du terminal.
Le fonctionnement de cette logique est assuré en partie par le matériel et en partie par le logiciel.

La partie matérielle est constituée de circuits dont les fonctions de transfert sont figées.

La partie logicielle est constituée du programme contenu dans la mémoire. Il donne les directives nécessaires au fonctionnement du matériel et peut être modifié (EPROM).
Le matériel,au fonctionnement rigide, réalise donc les séquences de base à caractère répétitif.
Seul le fonctionnement d'aspect matériel est développé dans ce chapitre.

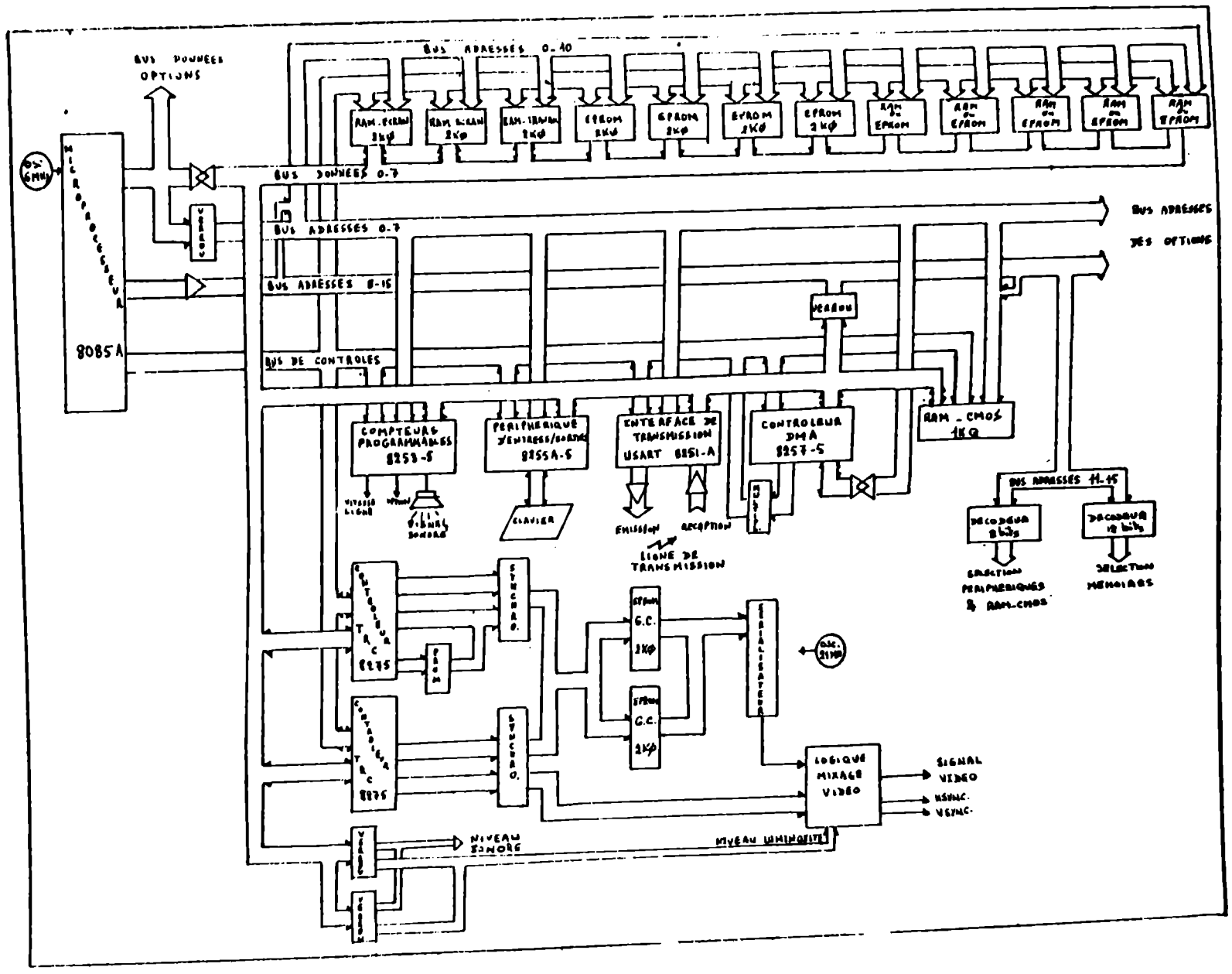


FIGURE 15 : SYNOPTIQUE DE LA LOGIQUE DE COMMANDE

1. ADRESSAGE DES MEMOIRES ET PERIPHERIQUES

Le microprocesseur fournit 16 bits d'adresse, correspondant à une capacité de 64 kilo-mots, plus un signal de sélection mémoire ou périphérique.

Le type d'adressage utilisé tient compte de la capacité de mémoire nécessaire à la logique de commande.
Parmi les quatre types de microprogramme qui sont prévus, le plus volumineux est celui de l'émulation du terminal VU 3 pour lequel les besoins en mémoire sont :

- 16 Kilo-octets pour le microprogramme de la carte de base.
- 4 Kilo-octets pour la mémoire d'écran (2000 caractères + 2000 attributs visuels).
- 2 Kilo-octets pour la mémoire de travail, soit 22 Kilo-octets pour la mémoire de base;
- 16 Kilo-octets pour l'extension mémoire
- 8 Kilo-octets pour l'option imprimante.
- 2 Kilo-octets pour l'option lecteur de badges + l'option lecteur de documents, soit 26 kilo-octets pour la mémoire optionnelle.

Finalement l'espace mémoire est partagé en 32 Kilo-octets pour la mémoire de base et 32 Kilo-octets pour la mémoire optionnelle.

La zone mémoire non affectée de la carte de base peut être utilisée pour la sélection des circuits périphériques, ceci permet au microprocesseur d'accèder aux périphériques comme à une mémoire. Les avantages de cette technique sont :

. Un coût en matériel moindre au niveau du décodage des sélections de circuits.

. Des performances en logiciel meilleures au niveau de l'exécution des instructions (accès mémoires plus rapides que les accès périphériques).

Les 8 Kilo-octets de l'espace mémoire libre permettent la réalisation simple d'un décodage de sélection de périphériques.

Les zones non affectées pour les cartes optionnelles doivent donner la possibilité d'adjonction de nouvelles cartes futures avec de nouvelles zones de mémoire. C'est pour cette raison que la zone d'adresse permettant la sélection des circuits est limité à 1 kilo-octet ; ceci entraîne un décodage d'adresse plus couteux que celui de la carte de base.

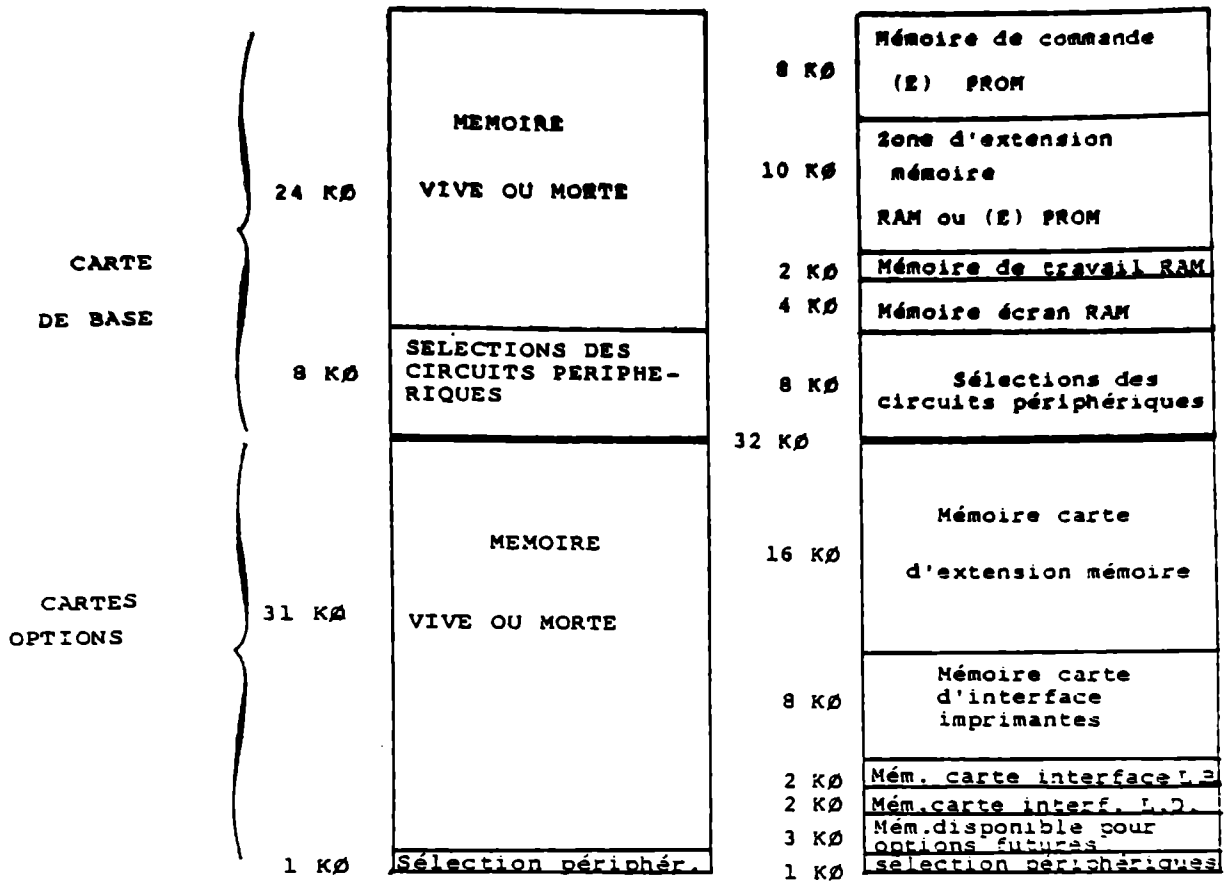
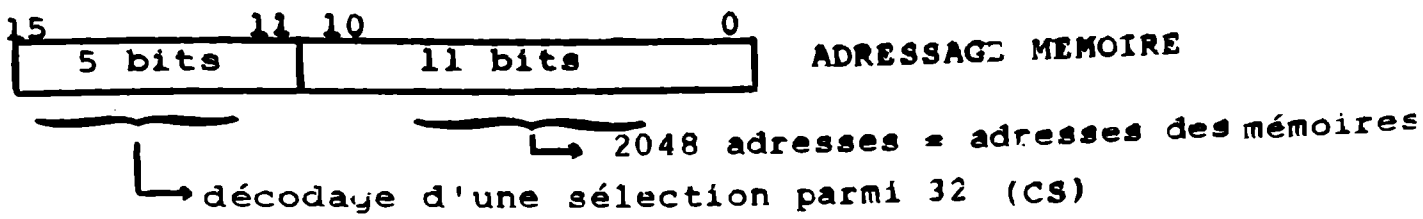


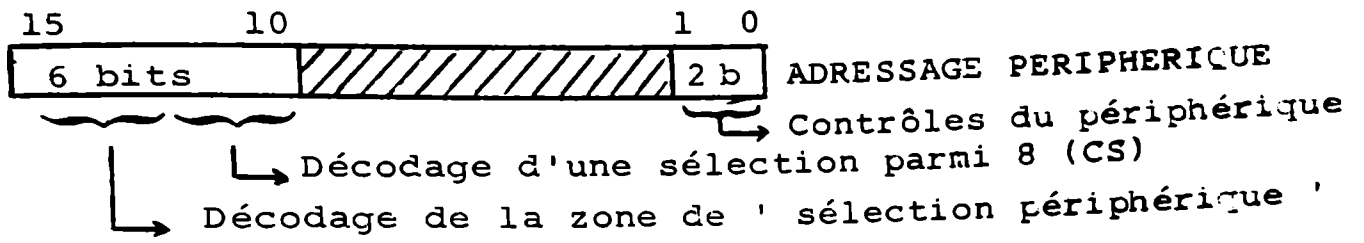
Figure 16 : REPARTITION DE L'ESPACE MEMOIRE (64 KØ)

Les mémoires dont la capacité est de 2 kø chacune ont un champ d'adresse découpé de la façon suivante :

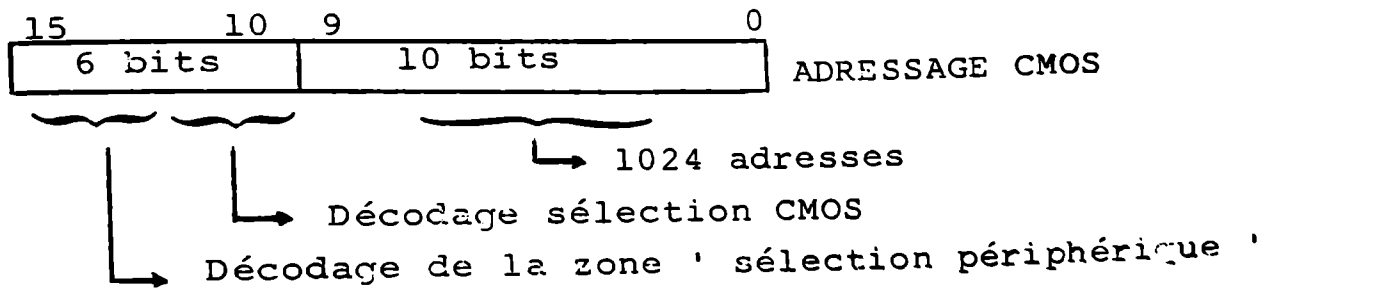


. <u>Adressages sur la carte de base</u> :

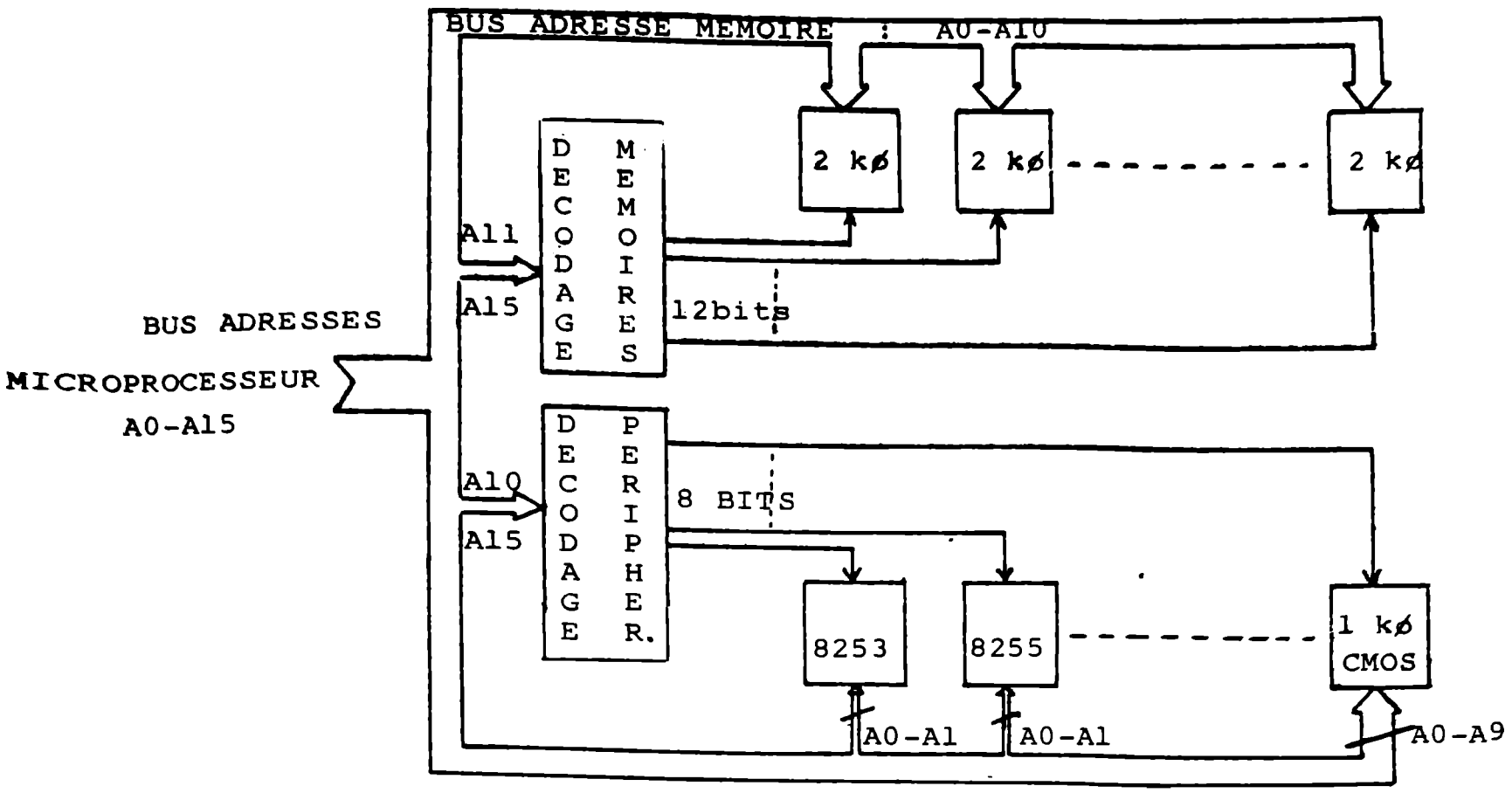
12 sélections sont affectées aux mémoires (12 x 2 = 24 kø)
4 sélections sont affectées aux circuits périphériques; ceux-ci sont au nombre de 7 mais puisque le champ d'adresse est limité à un ou deux bits de poids faibles, il suffit d'utiliser un bit supplémentaire (A10) pour le décodage de sélection.



La mémoire CMOS qui ne possède que 10 bits d'adresses (1Kx4) peut être adressée comme le huitième périphérique :

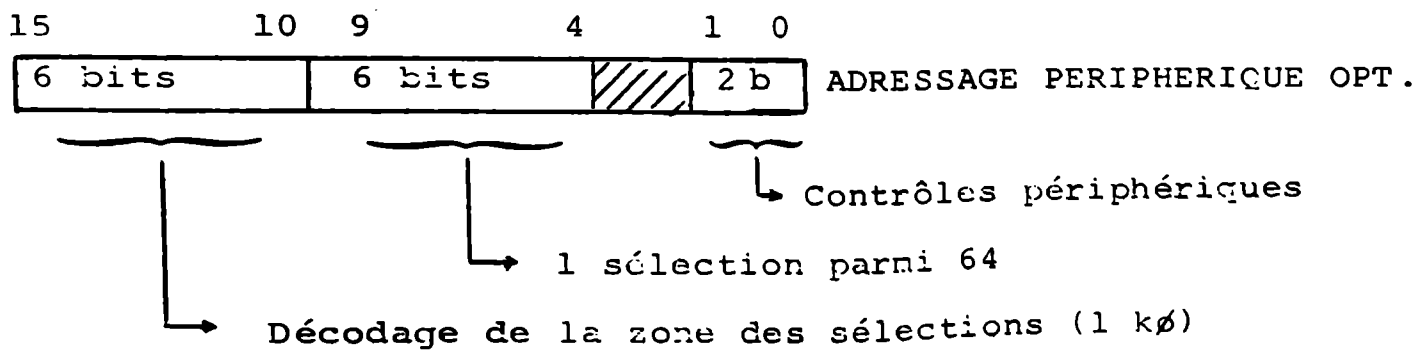


Trois décodeurs 8 bits (74 LS 138) permettent de générer les 20 sélections de circuits.



. Adressages sur les cartes optionnelles

Sur les cartes optionnelles le décodage de la mémoire est du même type que sur la carte principale.
Par contre la sélection des circuits périphériques s'effectue dans une zone mémoire plus restreinte et le nombre de sélections possibles a été fixé à 64 ce qui donne :



2. TRANSFERT DE L'INFORMATION

A la mise sous tension de l'appareil, un circuit RC et un - trigger de Schmitt - interne au microprocesseur réalisent un effacement initial des circuits programmables, des bascules et des registres. Le registre ' compteur programme'du microprocesseur pointe ensuite la première adresse du microprogramme (mémoire de commande). Le déroulement du programme s' effectue à partir de lectures et d'exécution des instructions contenues dans la mémoire de commande. Ces instructions sont de deux types :

- Instructions à exécution interne au microprocesseur ; (exemple : addition de deux registres).
- Instructions à exécution nécessitant un échange (lecture ou écriture) avec un circuit mémoire ou périphérique ; (ex. : chargement du contenu d'un registre du microprocesseur dans une case mémoire).

Ces opérations de programme sont réalisées à partir de signaux de base du microprocesseur dont les cycles fondamentaux sont :

- Le cycle de recherche d'une instruction en mémoire de commande (lecture)
- Le cycle d'exécution d'une lecture de mémoire ou de périphérique.
- Le cycle d'exécution d'une écriture de mémoire ou de périphérique.

La logique réalisée utilise un second processeur, le DMA, esclave du 8085A mais qui vient se substituer à lui pour réaliser dans une même séquence :

' 1 lecture + 1 écriture ' ou bien ' 1 écriture + 1 lecture '.

C'est donc à partir des séquences de base du microprocesseur et du DMA que sont réalisés tous les transferts d'information entre les circuits de la logique de commande.

2.1. Types de transferts

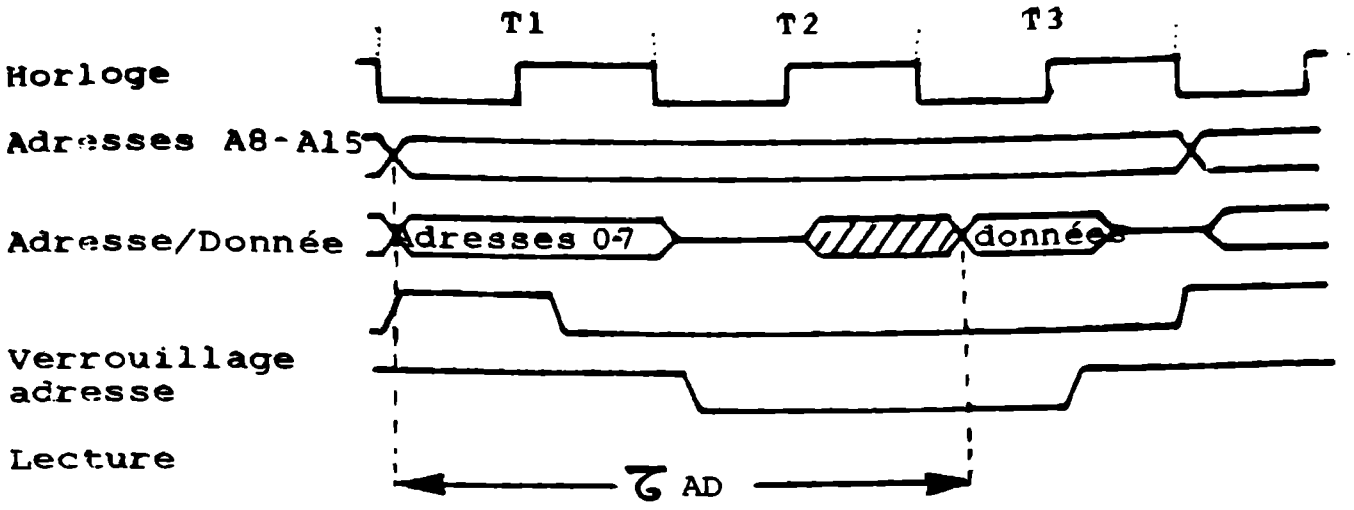
2.1.1. Transferts gérés par le microprocesseur 8085 A

Microprocesseur	⟵	Mémoires EPROMS
Microprocesseur	⟷	Mémoires RAMS
Microprocesseur	⟷	Contrôleur DMA (8257-5)
Microprocesseur	⟷	Interface clavier (8255A-5)
Microprocesseur	⟷	Interface de communication (8251A)
Microprocesseur	⟷	Compteurs programmables (8253-5)
Microprocesseur	⟷	Contrôleur de CRT (8275-2)
Microprocesseur	⟶	Verrous (8282)

2.1.2. Transferts gérés par contrôleur DMA :

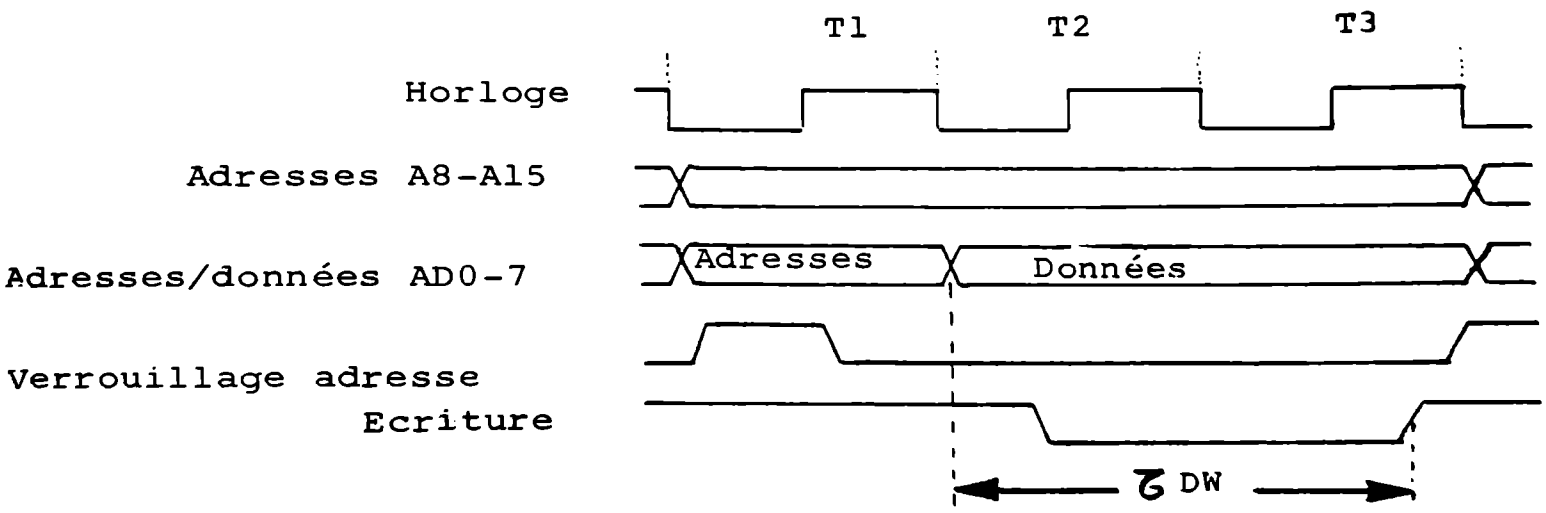
Mémoire RAM	⟷	Interface de communication (8251A)
Mémoire d'écran	⟶	Contrôleur de CRT (8275-2)

2.2. Cycles de lecture et d'écriture microprocesseur



- Cycle de lecture

Le cycle de lecture s'effectue sur trois cycles d'horloge soit 3 x 325 = 975 ns.



- Cycle d'écriture

Le cycle s'effectue sur trois cycles d'horloge, soit 3 x 325 = 975 ns.

Les temps de cycles générés par le microprocesseur Intel sont évidemment adaptés pour la connexion des circuits périphériques du même fournisseur.
Toutefois l'introduction de couches logiques entre les circuits, l'augmentation de charges capacitives sur les bus par la connexion de nombreux circuits et l'utilisation de mémoires aux temps d'accès les plus lents possibles pour des raisons de coûts, nécessitent un calcul des temps des transferts sur les chaînes les plus critiques.

. L'étude montre que le temps le plus critique lors d'une lecture est τ_{AD}, délai entre l'apparition des signaux d'adressage à la sortie du microprocesseur et l'arrivée des données en provenance de circuit lu.

La condition à satisfaire est :

$$\tau = [tp1 + tc1 + tacc + tp2 + tc2]_{maximum} < \tau_{AD\ maximum}$$

avec tp1 = temps de propagation sur le bus 'adresse'

tc1 = retard dû aux capacités sur le bus 'adresse'

tacc = temps d'accès du circuit lu

tp2 = temps de propagation sur le bus données

tc2 = retard dû aux capacités sur le bus données.

Application au cas le plus défavorable: la lecture dans la mémoire CMOS :

$\tau_{AD\ maximum} = (\frac{5}{2} T - 225) = 587$ ns

tp1 max. = 92 ns

tp2 max. = 18 ns

tacc max. = 450 ns

tc2 max. = 36 ns (tc1 ≠ 0)

Le temps τ = 596 ns > τ_{AD}, peut être pondéré suivant une règle statistique de technologie qui ramène le temps maximum τ à un temps maximum τ' plus proche de la réalité τ' = 570 ns < 587 ns.

. L'étude montre que le temps le plus critique lors d'une écriture est τ DW, durée de stabilité des données à l'entrée d'un circuit avant l'écriture (front arrière du signal écriture).

La condition à satisfaire est :

$$\tau = [tpw + t1w]\ \text{minimum} - [tp2 + tc2]\ \text{maximum} > \tau DW\ \text{minimum}$$

avec : tpw = temps de propagation du signal d'écriture.

t1w = largeur du signal d'écriture

tp2 = temps de propagation des données

tc2 = retard dû aux charges capacitives sur le bus données.

Application au cas le plus défavorable : l'écriture dans le circuit de compteurs programmables 8253-5.

τ Dw minimum = 250 ns

tpw mini $\approx$ 0 ns

t1w mini = $\left(\frac{3}{2}\ T - 80\right)$ = 407 ns

tp2 max. = 18 ns

tc2 max. = 13 ns

d'où :

τ = 376 ns > 250 ns

2.3. <u>Cycles de lecture et d'écriture DMA</u> :

Le circuit DMA est utilisé pour des transferts directs de périphérique à mémoire (cas de la réception de blocs de données sur l'interface de communication) ou de mémoire à périphérique (cas de l'émission de blocs de données vers l'interface de communication et de l'émission de bloc de caractères de la mémoire d'écran vers le contrôleur de CRT).

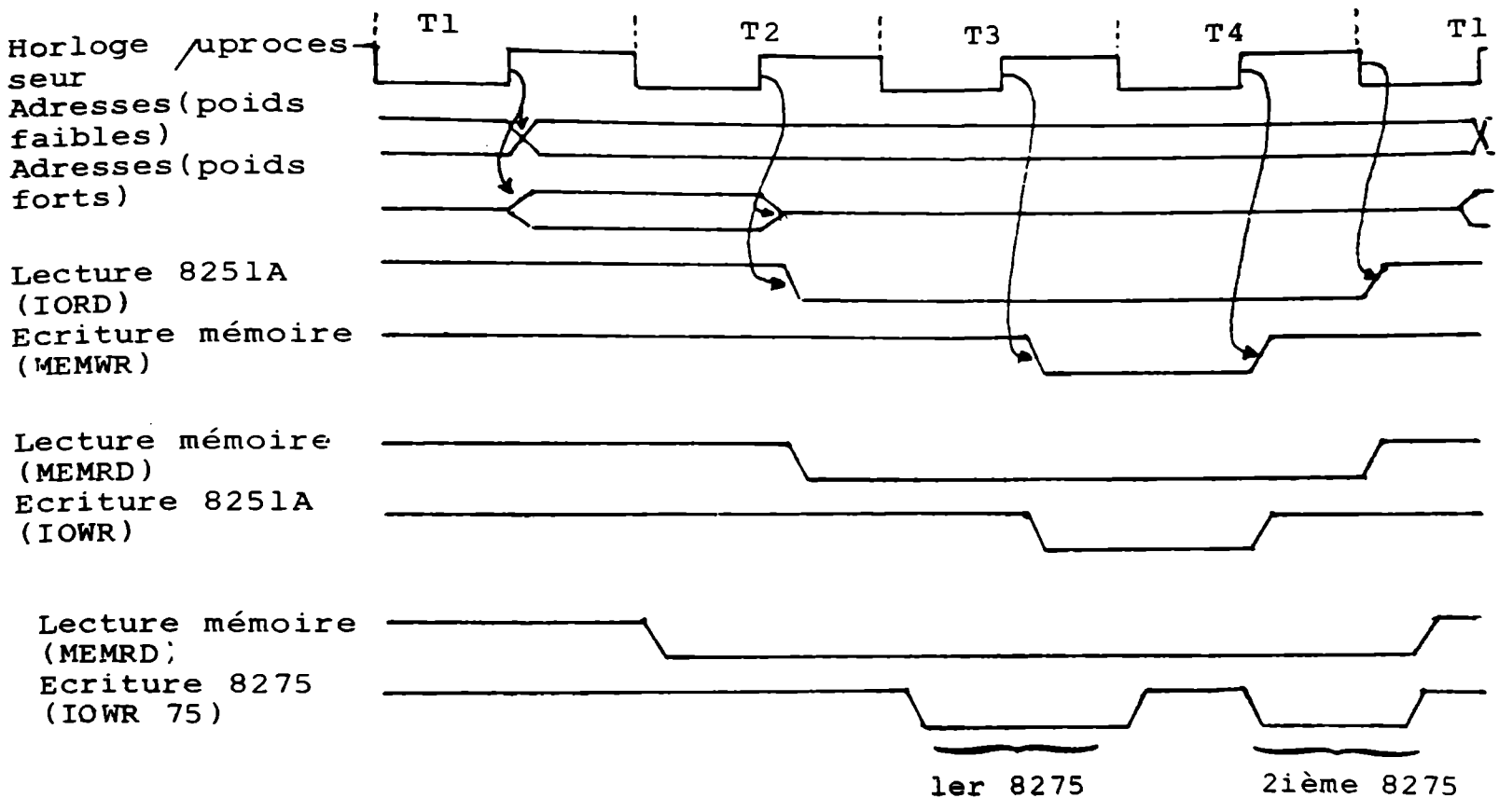
Le circuit,qui possède quatre canaux de transferts,gère plusieurs transferts en ' temps partagé '.

Les canaux sont affectés de la façon suivante :

canal 0	:	réception de données sur l'interface de transmission
canal 1	:	émission de données vers l'interface de transmission
canaux 2 et 3	:	émission de données vers les contrôleurs de CRT.

L'étude à montré l'utilité de modifier les cycles de transfert vers les contrôleurs de CRT afin d'effectuer deux écritures dans un même cycle DMA.

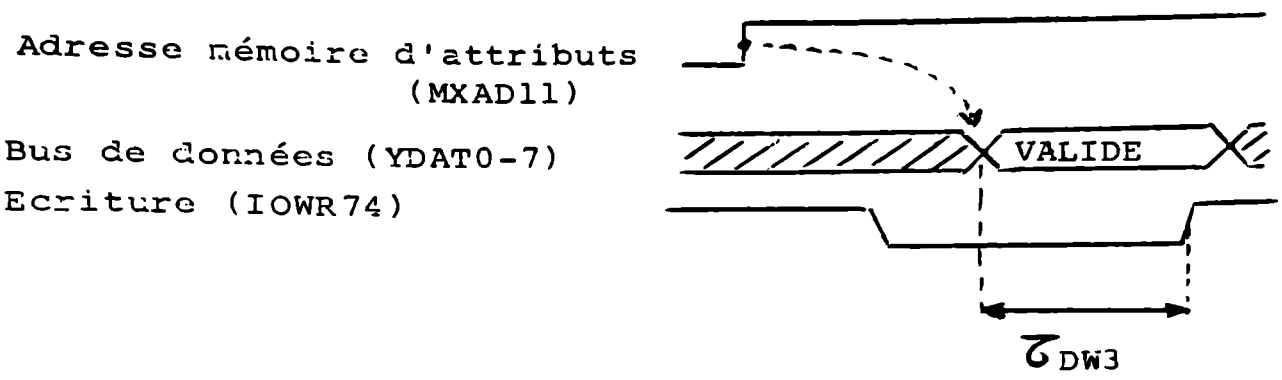
Les trois types de cycles utilisés sont définis par les signaux suivants :



L'étude des chronogrammes (annexes) montre que seul le troisième type de cycle présente des temps critiques. Le cas le plus défavorable est celui de l'écriture dans le second contrôleur 8275 pour lequel le temps τ DW3 (durée de stabilité des données à l'entrée du circuit avant l'écriture) doit satisfaire à la condition :

$$\tau_{DW3} > \tau_{DW\ mini\ du\ 8275}$$

$$\tau_{DW\ minimum} = 150\ ns$$



$$\tau_{DW3} = [tpw + tcw]\ minimum - [tP1 + tcl + tacc + tP2 + tc2]_{max}$$

avec
- tpw = temps de propagation du signal écriture = 490 ns
- tcw = retard dû aux charges capacitives sur le signal écriture $\approx$ 0 ns
- tpl = temps de propagation de MXAD11 = 105 ns
- tcl = retard dû aux charges capacitives adresse $\approx$ 0ns
- tacc = temps d'accès mémoire 'écran attribut' = 200 ns
- tp2 = temps de propagation bus de données $\approx$ 0 ns
- tc2 = retard dû aux charges capacitives sur le bus de données = 35 ns

Finalement

$$\tau_{DW3} = 490 - 340 = 150 \text{ ns minimum}$$

$$\tau_{DW3} = 150 \text{ ns} \geq \tau_{DW \text{ mini}} (150 \text{ ns})$$

En résumé, il faut rappelerque l'utilisation d'un contrôleur DMA permet de ramener le temps de transfert d'un octet entre une mémoire et un circuit périphérique de 9 µs, traité par le microprocesseur, à 1,3 µs. L'utilisation du DMA n'est efficace que pour des blocs de données dont les adresses de rangement en mémoire doivent être contigües.

L'étude des chronogrammes des différents cycles de transfert fait apparaitre que :

- Les mémoires ont été choisies avec les temps de traversée les plus lents possibles pour des raisons de coût.
- L'adjonction de quelques couches logiques entre le microprocesseur et un circuit périphérique rend rapidement critiques les temps de propagation des signaux.
- La charge capacitive sur le bus de données devient assez pénalisante lorsque de nombreux circuits y sont connectés.

3. LA VISUALISATION

Le choix du type de visualisation a permis de décrire dans le principe le fonctionnement de la visualisation des caractères de l'écran.

Ce chapitre se limite donc à une description technique des principales fonctions de la visualisation.

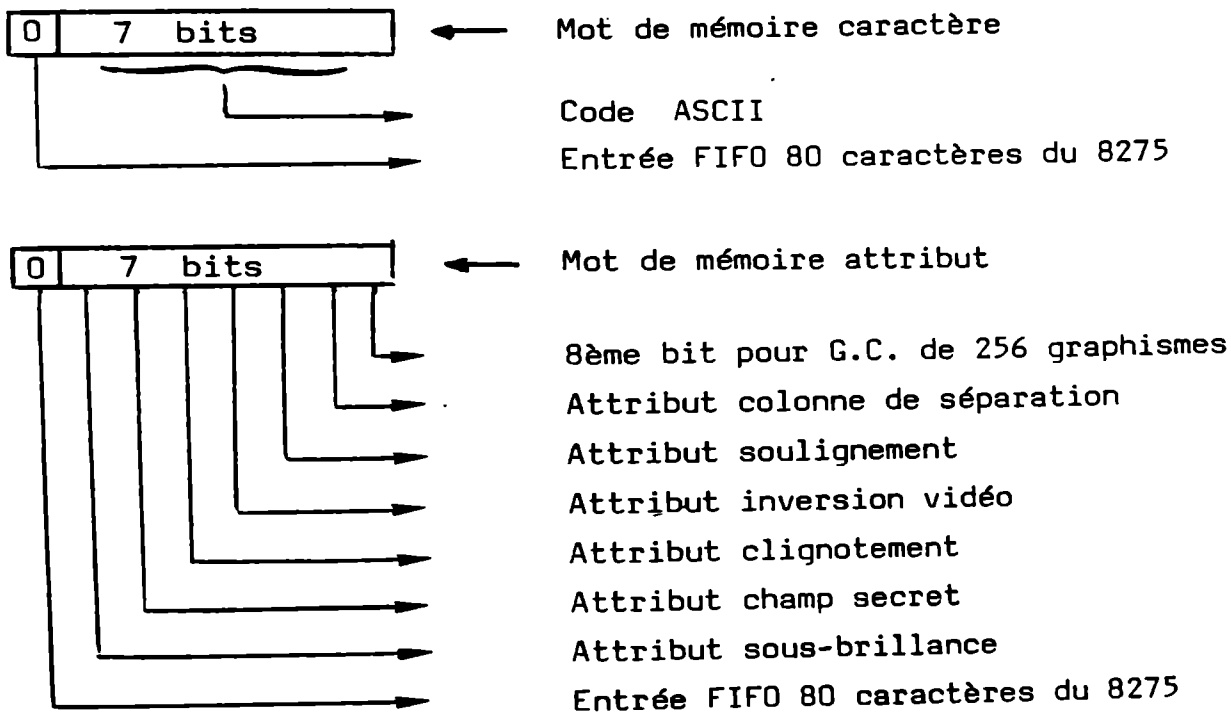
3.1. Lecture et écriture en mémoire d'écran

Les codes reçus sur la ligne de communication sont :

- soit des codes de commandes
- soit des codes de caractères

Les codes de commandes sont traités, et s'il s'agit de commandes d'attributs visuels, ceux-ci sont rangés dans la mémoire d'attributs à l'emplacement correspondant à la position courante du curseur.

Les codes de caractères sont rangés dans la mémoire de caractères à l'emplacement correspondant à la position du curseur.



Les codes écrits dans la mémoire d'écran peuvent aussi provenir du clavier ou d'un lecteur (lecteur de badges par exemple). La lecture de la mémoire d'écran s'effectue soit à partir du microprocesseur ou du DMA pour l'émission vers la ligne de transmission (données envoyées vers l'unité centrale), soit à partir du microprocesseur pour l'émission vers l'interface imprimante (recopie d'écran) ; enfin elle a lieu à partir du DMA pour le rafraîchissement de l'écran.

3.2. Rafraichissement de l'image

Le fonctionnement a été décrit précédemment (chapitre B 3.1) . La figure 17 représente le séquencement des informations au travers de la logique de visualisation.

La mémoire d'écran est entièrement parcourue en 20 ms (à 50 Hz) avec un temps d'occupation du bus 13 %. Une programmation judicieuse des contrôleurs de CRT permet, sans avoir à modifier le matériel, d'effectuer si nécessaire, le rafraîchissement avec une fréquence de 60 Hz (cas d'une alimentation secteur 60 Hz).

Pour cela, l'écran doit être rafraîchi plus rapidement (+ 20 %) ce qui s'obtient en ne balayant plus que 12 des 15 lignes de la rangée de caractères (suppression de 3 des 4 interlignes horizontales entre chaque rangée de caractères).

3.3. Horloges de la visualisation

Les horloges de base sont :

. L'horloge ' point ' (période : intervalle de temps entre deux points visualisés) qui est utilisée par le sérialisateur

. L'horloge ' caractère ' (période : 10 périodes ' point ')qui rythme :

- le contrôleur de TRC
- les synchronisateurs
- le chargement parallèle du sérialisateur

Les comptages réalisés par le contrôleur sont générés à partir de l'horloge ' caractère ' et de paramètres programmables ; ce sont :

- le comptage de caractères visualisés par ligne (paramètre Nc1)
- le comptage de retours de balayage horizontal (paramètre Nc2)
- le comptage de lignes par rangée (paramètre N1)
- le comptage de rangées visualisées dans l'écran (paramètre Nr1)
- le comptage de retours de balayage vertical (paramètre Nr2)
- le comptage de la position curseur (paramètre C)

Ces comptages permettent de réaliser :

- les signaux de balayage horizontal et vertical pour le moniteur vidéo.
- les signaux de forçage du signal vidéo pour :
  - . les retours de balayage (forçage à 0)
  - . le curseur (forçage à 1)
  - . le rythme du clignotement (forçage à 0)
  - . la position du soulignement (forçage à 1)

Calcul de la fréquence de l'horloge ' Point '

La fréquence ' point ' Fp' idéale est donnée par la relation :

$$Fp' / Fi = (Nc1 + Nc2').\ Nl.\ (Nr1 + Nr2').\ Np$$

avec Fi : fréquence d'horloge image (50 Hz)

avec les caractéristiques de visualisation :

Np : nombre de points par segment = 10
Nc1 : nombre de caractères par ligne = 80
Nl : nombre de lignes par rangée = 15
Nr1 : nombre de rangées par écran = 25

Avec les temps de retours caractéristiques du tube (figure 18) :

11 /us pour le balayage horizontal ce qui correspond à :

$$Nc2' = \frac{11}{10\ Tp'}$$ /us (Nc2' exprimé en nombre de périodes 'caractère ')

2,14 ms pour le balayage vertical ce qui correspond à :

$$Nr2' = \frac{2{,}14 \cdot 10^3}{10\ Tp'\ (80 + Nc2')\ 15}$$ /us (Nr2' exprimé en nombre de périodes de rangée de caractère)

$$(Tp' = 1/Fp')$$

d' où :

$$Fp' = 50 \left[(80 + 1{,}1 \cdot 10^{-6}\ Fp') \cdot 15 \cdot \left(25 + \frac{14{,}26.10^{-6}\ Fp'}{80 + 1{,}1 \cdot 10^{-6} Fp'}\right)\right]$$

$$Fp' = 21{,}842 \text{ Mhz}$$

Toutefois, les paramètres Nc2 et Nr2 doivent être des nombres entiers, d'où :

$$Nc2' = 1{,}1 \cdot Fp' = 24{,}026 \qquad Nc2 = 24$$

$$Nr2' = \frac{2{,}14.10^3\ Fp'}{150\ (80 + 24{,}026)} = 2{,}995 \qquad Nr2 = 3$$

Finalement, la fréquence réelle est :

$$Fp = 50\ (80 + 24\ )\ 15\ (25 + 3\ )\ 10 = 21{,}840 \cdot 10^6 \text{ Hz}$$

Fp = 21,840 Mhz

A partir de la valeur Fp, les périodes des signaux de visualisation et les paramètres correspondants donnent :

- Période d'horloge ' point ' : Tp = 45,78 ns
- Période d'horloge ' caractère ' : Tc = 457,8 ns (Np = 10)
- Période d'horloge de synchronisation horizontale : TSYNC = 47,6 µs (Nc1 = 80)
- Période d'horloge rangée de caractères : Tr = 714,3 µs (N1 = 15)
- Période d'horloge de synchronisation verticale : TVSYNC = 20 ms (Nr1 = 25)

$$\text{Paramètre Nc 2} = \frac{11\ \mu s}{457,8\ ns} = 24$$

$$\text{Paramètre Nr2} = \frac{2,14\ ms}{714,3\ \mu s} = 3$$

3.4. Le curseur et les attributs visuels .

- La visualisation du curseur sur l'écran est entièrement gérée par le contrôleur de CRT. Des paramètres d'initialisation du contrôleur définissent un type de curseur, tiret souligné ou ' pavé ', clignotant ou non clignotant. La position du curseur sur l'écran est codée par une adresse rangée dans un registre. Pendant le rafraîchissement d'écran la coïncidence de cette adresse avec l'adresse du caractère rafraîchi provoque la génération du signal ' curseur ' (VSP/LTEN/RVV). La position du curseur est gérée par le microprogramme qui peut écrire le registre d'adresse curseur.

- Il a été démontré que, pour offrir la possibilité de générer autant d'attributs visuels qu'il y a de caractères, il faut utiliser deux contrôleurs de CRT. Le second contrôleur reçoit les codes des attributs visuels.

  Ce contrôleur génère à sa sortie un mot ' attribut ' de 6 bits en même temps que le premier contrôleur génère le code ' caractère ' associé. A chaque bit du mot correspond un attribut visuel.

Certaines fonctions de visualisation sont programmables et internes au contrôleur de CRT :

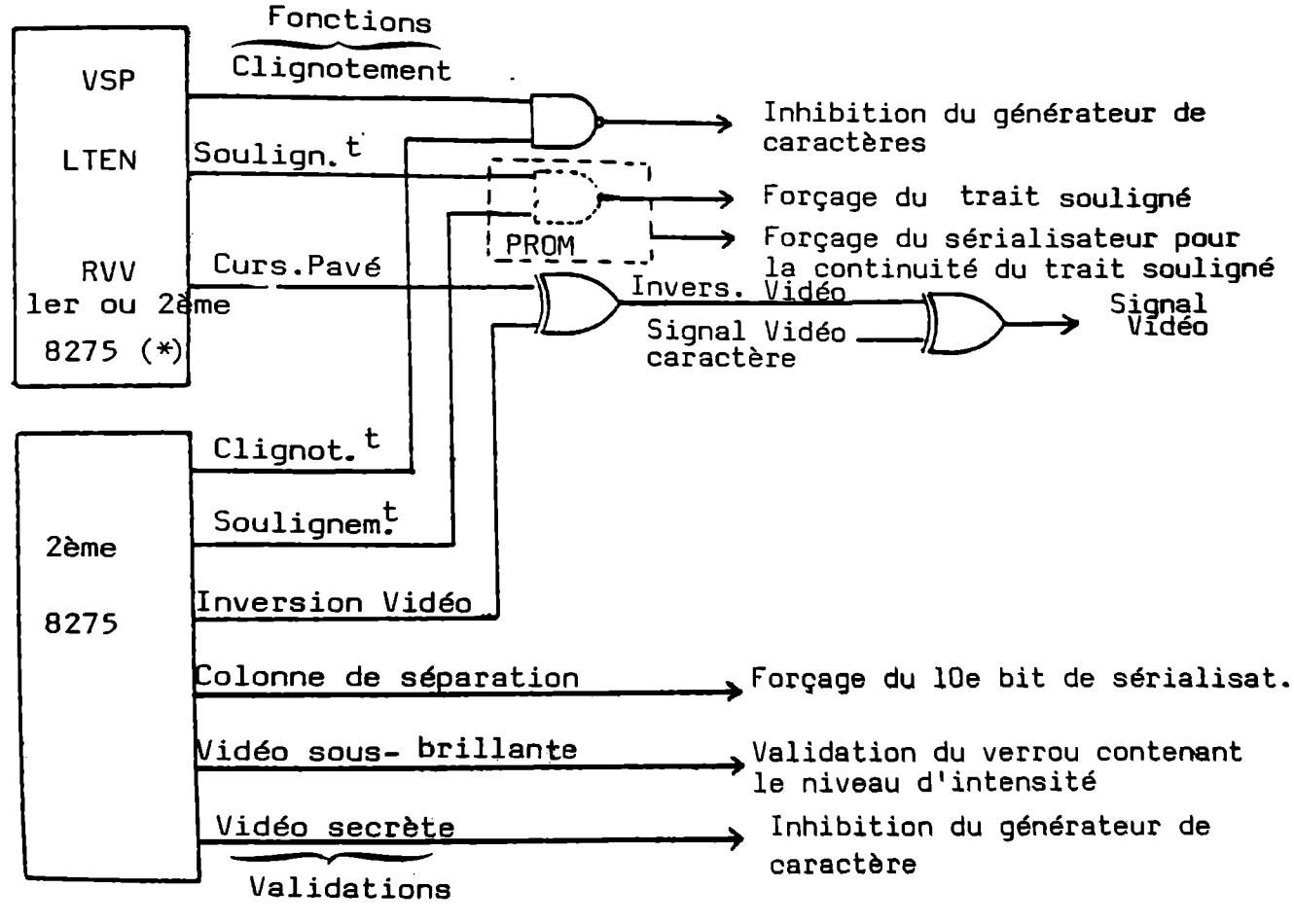
- la fonction de clignotement (inhibition du signal vidéo toutes les 0,64 secondes = $\frac{32}{50\ Hz}$)

- le décodage de la ligne de la matrice de caractères sur laquelle sont générés le curseur et le soulignement

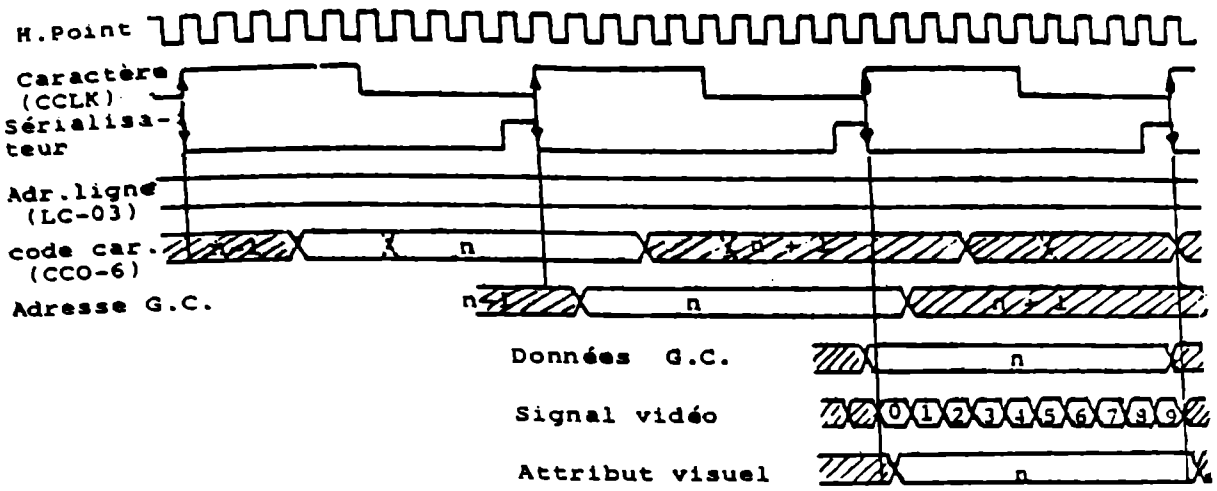
- le curseur type ' pavé ' en inversion vidéo

Le principe de génération des attributs visuels consiste à réaliser le ' ET LOGIQUE ' entre les signaux de génération des fonctions visuelles et les bits de validation du mot ' attribut '.

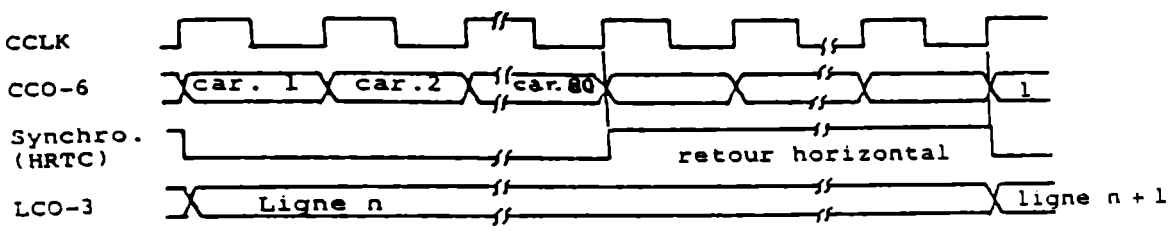
Des synchronisateurs sont nécessaires pour retarder les signaux d'attributs visuels afin qu'ils agissent, sur le signal vidéo, en phase avec les caractères auxquels ils sont associés.



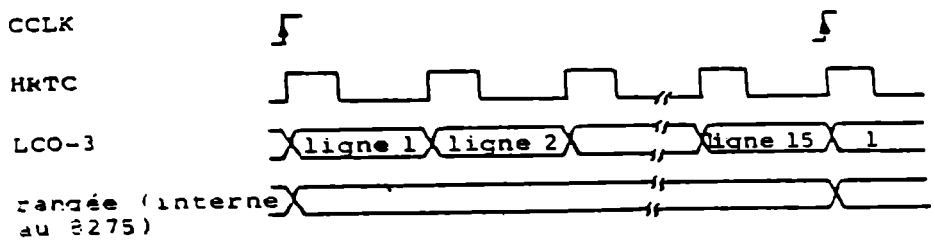
(*) Les deux contrôleurs sont programmés de façon identiques



TRANSFERT D'UN SEGMENT DE POINTS



TRANSFERT D'UNE LIGNE POINTS



TRANSFERT D'UNE RANGEE DE CARACTERES

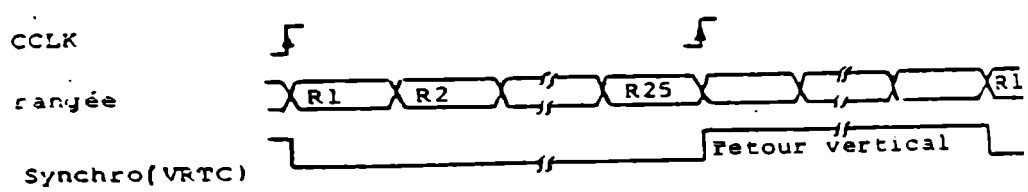
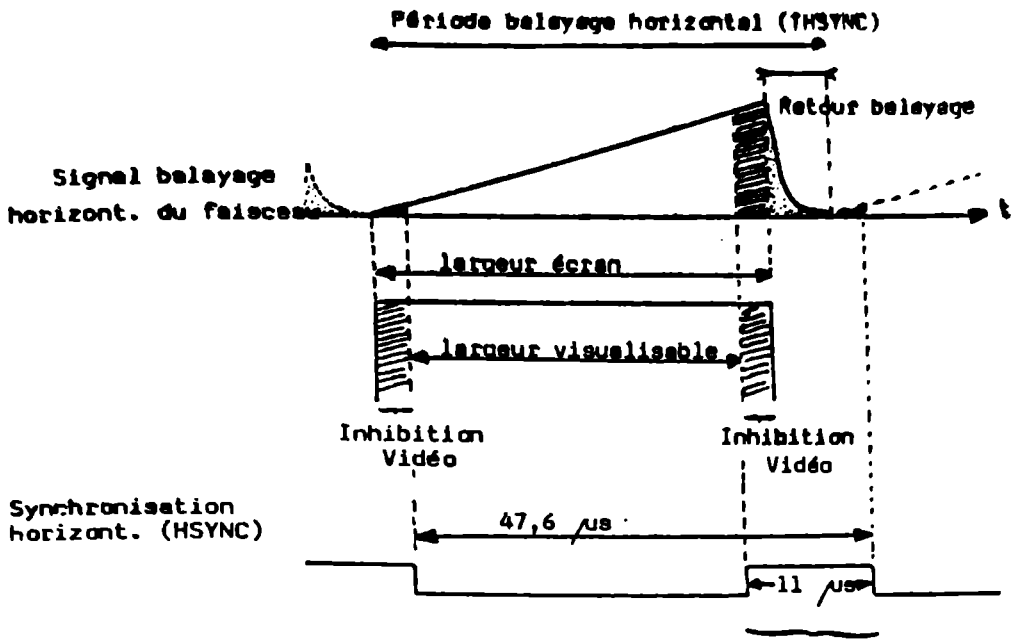


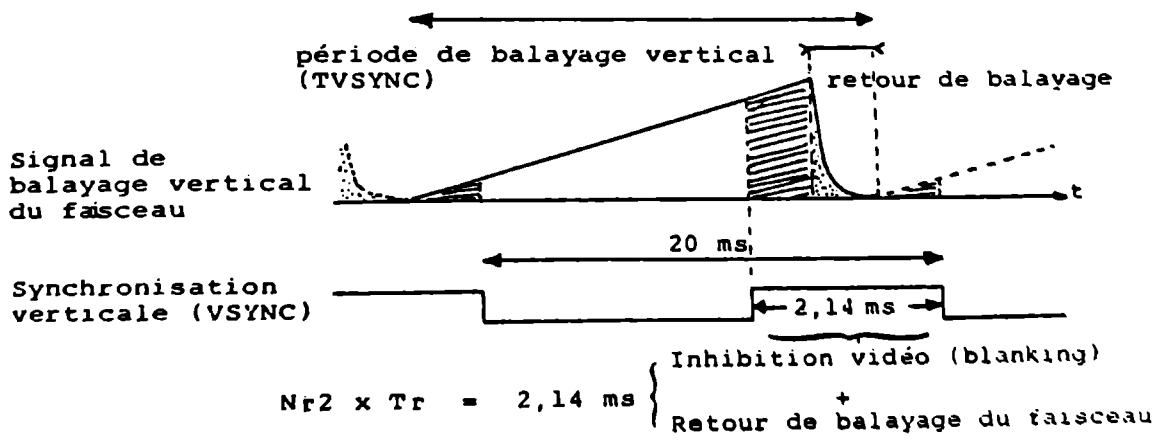
Figure 17 : TRANSFERT IMAGE ECRAN



Inhibition vidéo (blanking)
+
Retour de balayage du faisceau

Nc 2 x Tc = 11 µs

Figure 18-a : SIGNAL DE SYNCHRONISATION HORIZONTALE



Nr2 x Tr = 2,14 ms { Inhibition vidéo (blanking) + Retour de balayage du faisceau

Figure 18-b : SIGNAL DE SYNCHRONISATION VERTICALE

4. FONCTIONS NOUVELLES

La logique de commande qui réalise les fonctions nouvelles est de caractère plus original que fondamental ; seuls les principes de fonctionnement seront donc décrits.

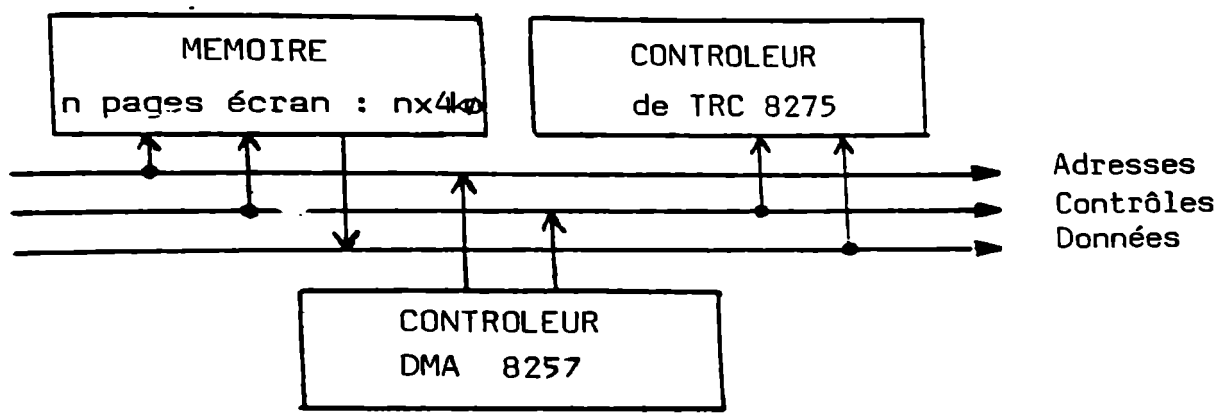
4.1. Le fonctionnement ' multi-page ' et le partitionnement d'écran

Le rôle de la fonction ainsi que les contraintes qu'elle a impliqué sur le choix du type de visualisation ont été précédemment développés.

Les circuits utilisés par la fonction sont :

- n fois 4 kø de mémoire RAM pour n page de mémoire écran
- le contrôleur DMA (8257)
- le contrôleur de CRT (8275)

Seule la mémoire d'écran supplémentaire est spécifique.



La fonction est gérée principalement par programme (firmware).

. La fonction ' multi-page '

Le transfert des données à partir du microprocesseur et le rafraîchissement d'image s'effectuent sur la page de mémoire sélectée de la même façon qu'avec un écran unique.

. La fonction partitionnement d'écran

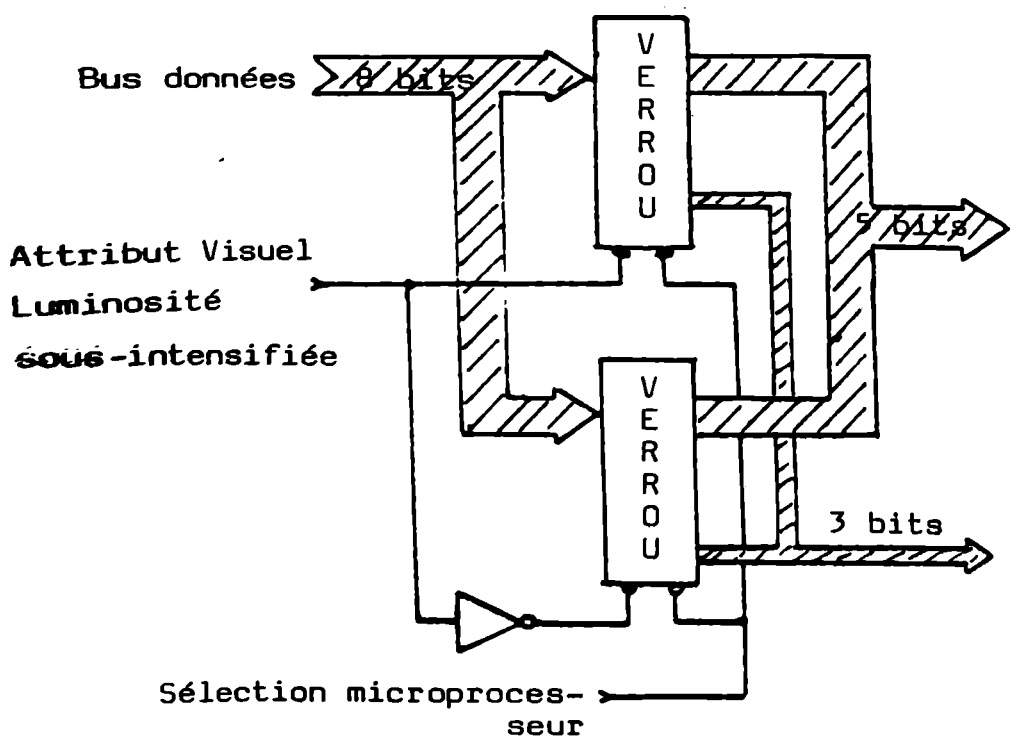
Sur commande reçue de la ligne de transmission, le microprogramme découpe la mémoire d'écran en plusieurs partitions, chacune correspondant à une zone d'écran. Le transfert des données (caractères et commandes) à partir du microprocesseur s'effectue dans la partition sélectée de la même façon que pour une mémoire d'écran normale . Le rafraîchissement nécessite une gestion particulière :

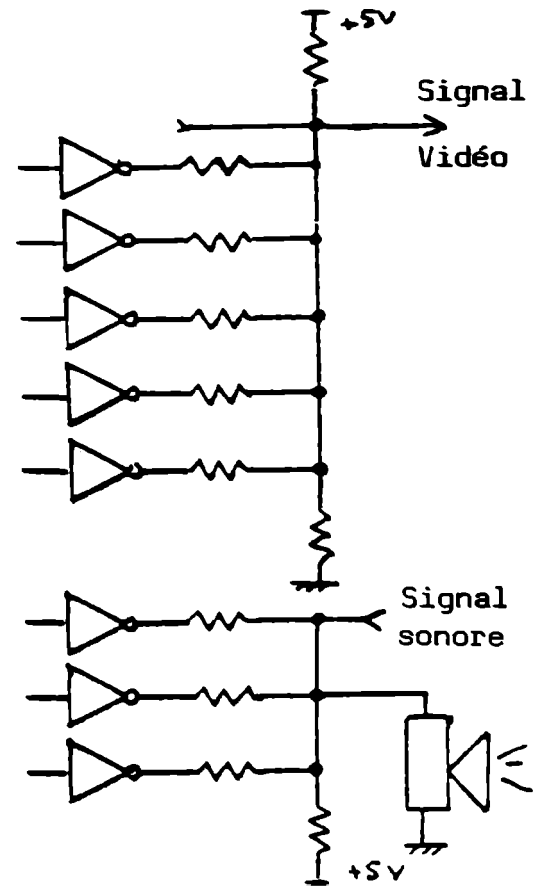
- l'interruption du microprocesseur, pour réinitialiser le DMA, ne s'effectue plus en fin d'écran mais à la fin de chaque partition.

- sur l'interruption, le microprocesseur range dans le DMA l'adresse mémoire écran du début de lapartition suivante ainsi que sa longueur en nombre d'octets.

4.2. Réglages des niveaux de luminosité et sonore

L'opérateur a la possibilité de régler la luminosité du faisceau d'écran ainsi que le niveau de l'alarme sonore à partir du clavier.





La séquence reçue du clavier est décodée et traitée ; deux octets correspondant aux niveaux de luminosité normale et sous-intensifiée sont chargés dans les verrous.

Le signal ' attribut ' valide l'un ou l'autre des deux verrous, suivant l'état luminosité normale ou réduite.

Les 5 bits du code de niveau, appliqués sur un pont diviseur de tension, fournissent 32 niveaux de luminosité.

Le signal sonore est réglé suivant le même principe (sans notion d'attribut) sur 8 niveaux d'intensité (3 bits).

4.3. Sauvegarde de la configuration du terminal

La sauvegarde des informations de configuration du terminal qui sont accessibles à l'opérateur (sélections techniques, réglages de luminosité , réglages sonores ...) est réalisée à partir des 3 éléments (figure 19) :

- un détecteur de niveau de tension
- une mémoire vive
- une batterie

. Lorsque la tension d'alimentation (+ 5 Volts) est établie la diode Zener (VZ) polarise la tension en A à un niveau qui rend T3 et T4 passants. Le transistor T1 est saturé et la mémoire est alimentée par Vcc = 5 volts - Vce sat.

La tension en B est polarisée par un diviseur de façon à ce que T2 soit passant et que le signal ' VALID ' soit à un niveau logique'1' (2 volts $<$ Vv $<$ 5 Volts).

La batterie de tension 2,4 volts est alors polarisée en inverse et se recharge.

. Lorsque la tension d'alimentation (5 volts) descend et qu'elle atteint 4,75 volts, la tension Va ($<$ Vbe passant) bloque T3 et T4 et par conséquent T1 et T2 ; c'est la batterie qui alimente alors la mémoire à 2,4 volts. Le signal ' VALID ' forcé à ' 0 ' bloque les signaux de lecture et d'écriture afin d'éviter des impulsions parasites pendant les transitoires.

- La mémoire RAM (type TC 5514 P de Toschiba) est conçue pour ce type d'application. Sa technologie C-MOS permet une très faible consommation en courant au repos. La tension d'alimentation au repos est comprise entre 2 et 5,5 Volts.

- Lorsque la batterie alimente la mémoire, elle génère du courant sur trois entrées (VCC, WE et CS) soit :

3 x 20 µA = 60 µA maximum

et 3 x 0,2 µA = 0,6 µA typique

d' où une autonomie d'alimentation donnée par le fabricant à 20°C
de : 25 jours, minimum
150 jours typique

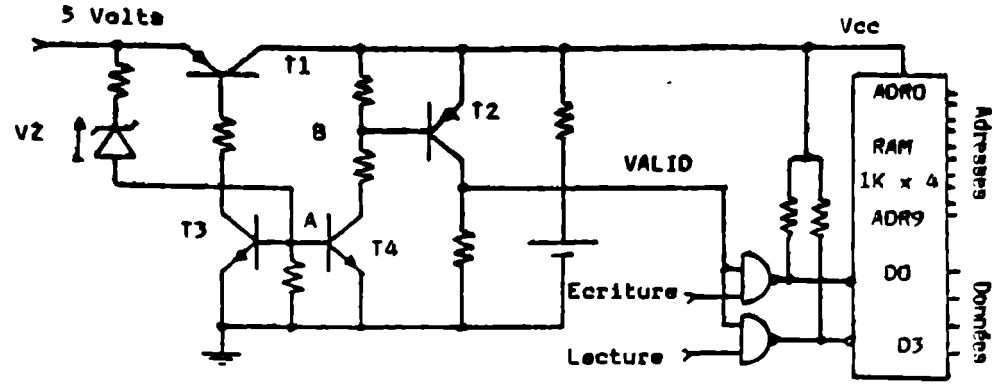


Figure 19-a SCHEMA DE PRINCIPE DU SYSTEME DE SAUVEGARDE

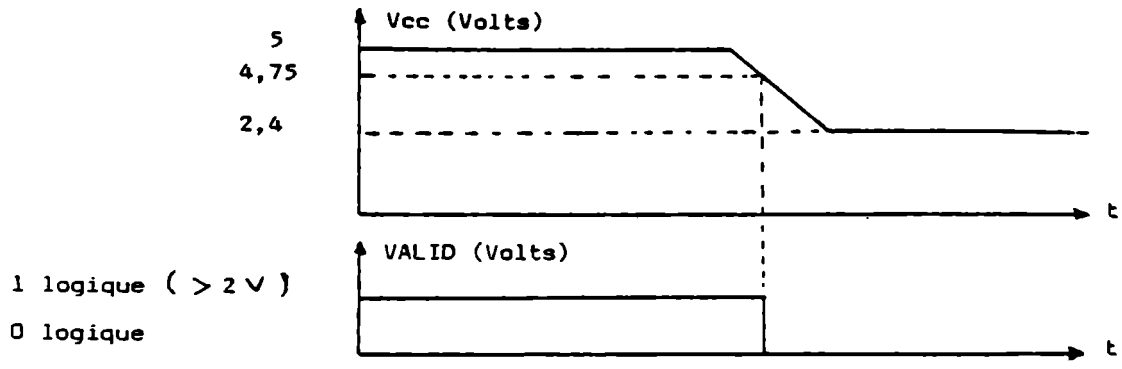


Figure 19-b : COMMUTATION D' ALIMENTATION

CONCLUSION

- En résumé, il faut rappeler que la logique de commande réalise d'une part le remplacement des terminaux existants,d'autre part les fonctions nouvelles permettant d'améliorer le produit, enfin qu'elle a été conçue de façon à permettre une extension future de ses possibilités.

  Le choix du découpage des cartes optionnelles pour les interfaces périphériques rend ceux-ci modulables et permet de les panacher suivant les besoins de l'utilisateur. Le coût du terminal a ainsi été optimisé.

  La logique de visualisation assure l'affichage de 2000 caractères sur un écran de type ' 15 pouces ' sans limitation du nombre des attributs visuels comme c'est le cas des terminaux actuels. Le nombre de graphismes de caractères a été doublé. Le réglage de la vidéo, la meilleure définition de la matrice du caractère, la sauvegarde d'informations à la mise hors tension apportent des améliorations d'ordre ergonomique. Enfin l'architecture de cette logique permet au programme de réaliser toutes les fonctions nouvelles et notamment le ' multi-page ' et le 'partitionnement d'écran'. Une extension de la mémoire sur une carte optionnelle est aussi possible pour ajouter des fonctions futures.

  Les performances (débits des informations sur les interfaces et rafraîchissement d'image) ont été maintenues malgré l'augmentation du degré d'intelligence.

  Le choix des composants, mémoires et LSI principalement, permet de classer le terminal dans la gamme des appareils à bas prix.

- L'étude réalisée a permis de retenir certains enseignements. Ainsi, la réalisation des interfaces avec les appareils périphériques a mis en évidence un manque de normalisation de la transmission pour les nombreux types d'appareils. Seuls des circuits entièrement programmables permettent actuellement d'adapter une même carte d'interface à plusieurs types d'appareils périphériques.

  La réalisation de la logique de visualisation a fait apparaitre que, pour améliorer les performances ou pour assurer des fonctions nouvelles, les circuits LSI disponibles sur le marché ne sont pas toujours adaptés et nécessitent de la logique complémentaire en circuits MSI. La solution qui consiste à faire fabriquer par le fournisseur des circuits ' à la demande ' impose des délais et des coûts trop élevés pour faire face à la concurrence.

Il faut donc souligner que la réalisation d'un ensemble logique à base de circuits LSI spécialisés, entraîne une certaine dépendance vis à vis des fabricants.

- Dans l'avenir, la réalisation de la logique de commande pour un terminal de visualisation devrait s'accomplir avec des circuits très standardisés et encore plus intégrés (pour la logique de visualisation par exemple). L'augmentation constante du volume du microprogramme qui tend à limiter les performances à cause du temps d'exécution des fonctions de plus en plus long, et l'augmentation des débits de transmission sur les lignes devraient entraîner prochainement l'utilisation de microprocesseurs à 16 bits de données.

ANNEXE (1)

FONCTIONS DU TERMINAL

FONCTIONS D'AFFICHAGE

	VU1	VU2	VU3 VU4
Fonctions de format			
Recul curseur d'un pas	X	X	X
Retour curseur au début de ligne	X	X	X
Passage curseur à la ligne suivante	X	X	X
Adressage absolu du curseur	X	X	X
Sélection des attributs visuels	X	X	X
Nouvelle ligne vers le bas d'écran	X		X
Nouvelle ligne vers le haut d'écran	X		
Fonctions d'édition			
Positionnement du curseur au début d'écran	X	X	X
Avance curseur	X	X	X
Recul curseur	X	X	X
Remontée curseur	X	X	X
Descente curseur	X	X	X
Positionnement curseur (X, Y)	X	X	X
Envoi position curseur	X	X	
Tabulation avant		X	X
Tabulation arrière		X	X
Pose effacement de taquets de tabulation			X
Insertion / suppression caractère			X
Insertion / suppression ligne			X
Fonctions d'effacement			
Effacement partiel ou total d'écran	X	X	X
Effacement partiel ou total de ligne	X	X	X
Effacement de champs			X

Fonctions de contrôle

	VU1	VU2	VU3 VU4
Signal sonore	X	X	
Définition d'un mode de fonctionnement	X	X	
Annulation d'un mode de fonctionnement	X	X	
Signalisation de l'état de l'appareil	X	X	X
Transfert de données à appareil périphérique	X	X	
Ré-initialisation microprogramme	X	X	X
Inhibition curseur	X		
Validation de saisies (clavier, badge,...)		X	
Inhibition de saisies		X	
Codage de l'octet fin de message	X	X	
Configuration de l'appareil	X	X	X
Début de zone fixe			X
Début de zone variable			X
Champ répétitif			X

FONCTIONS D'IMPRESSION

Saut de ligne	X	X	X
Saut de page	X	X	X
Retour chariot	X	X	X
Validation impression	X	X	X
Inhibition impression	X	X	X
Recopie page	X	X	X
Programmation longeur de page impr.	X	X	X
Impression en mode transparent	X	X	X
Répétition de caractères tampons			X
Impression de champs fixes			X
Impression de champs variables			X

NOUVELLES FONCTIONS

Sélection jeu de caractères

Attente de message

Commande de voyants clavier

Codage de touches de fonction du clavier

Répétition d'un code

Passage en ' minuscule '

Passage en ' majuscules '

Suspension de réception ligne

Reprise de réception ligne

Passage en mode de transmission

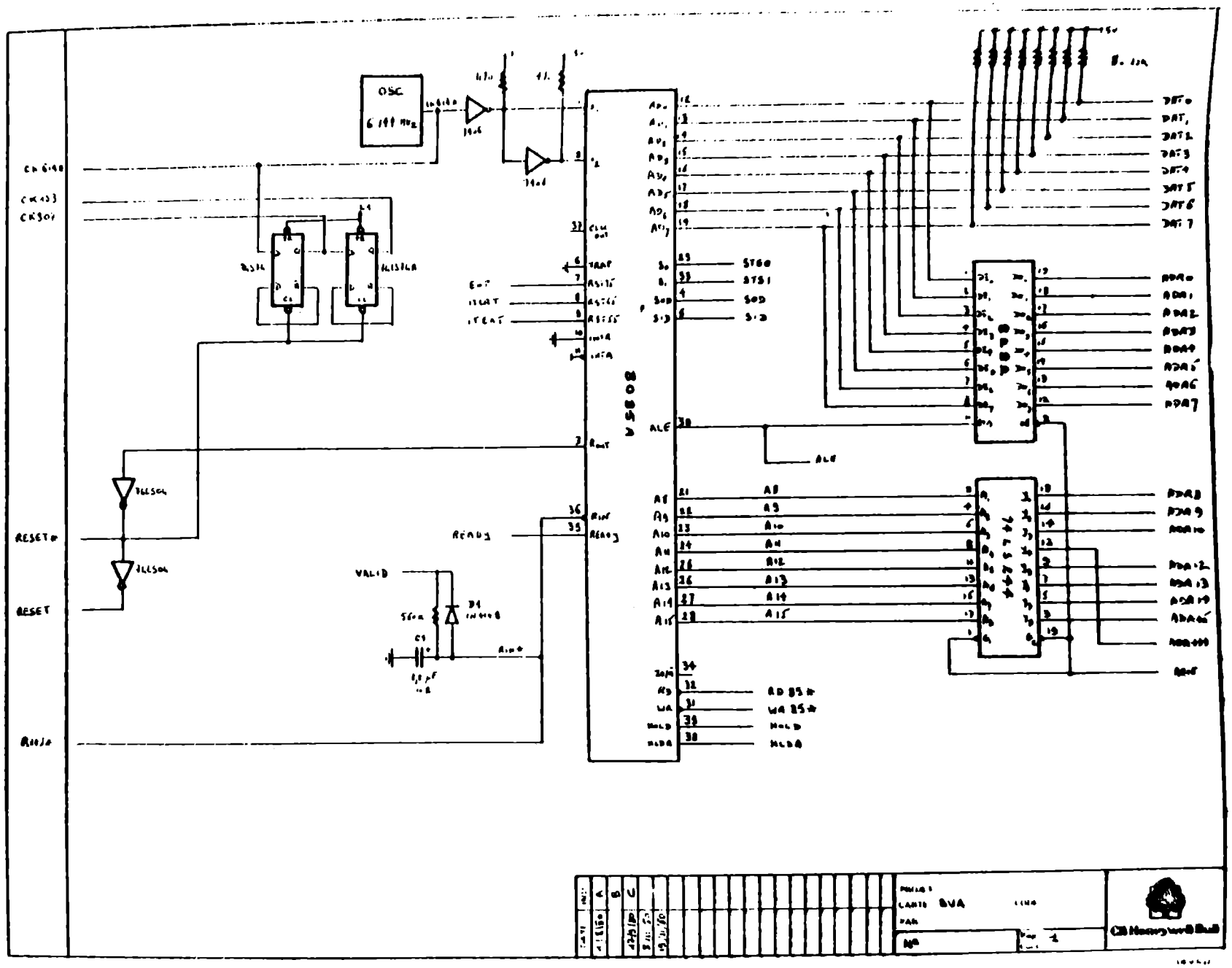
Partitionnement de la mémoire écran

Sélection de partition de la mémoire écran

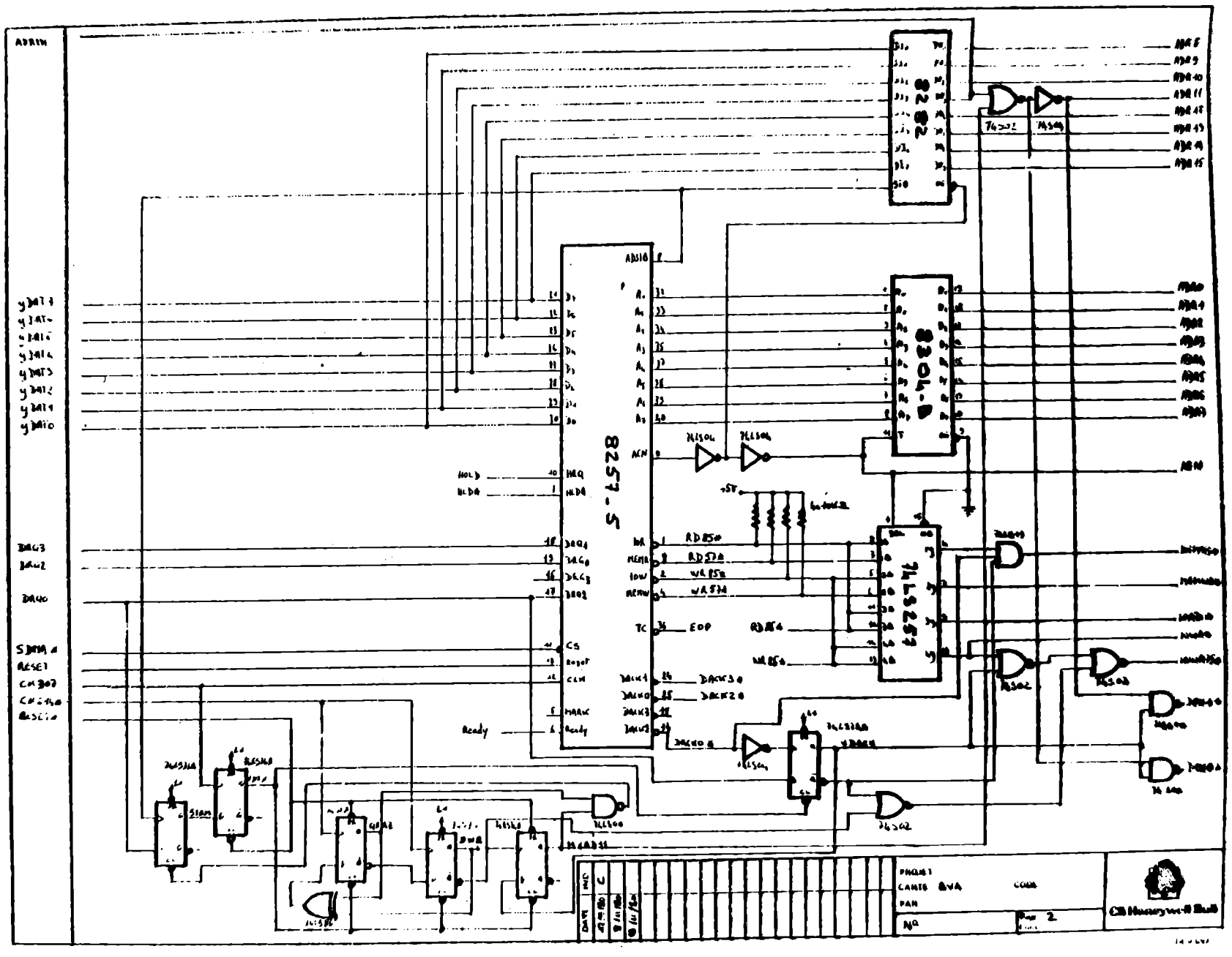
Partitionnement de l'écran.

ANNEXE (2)

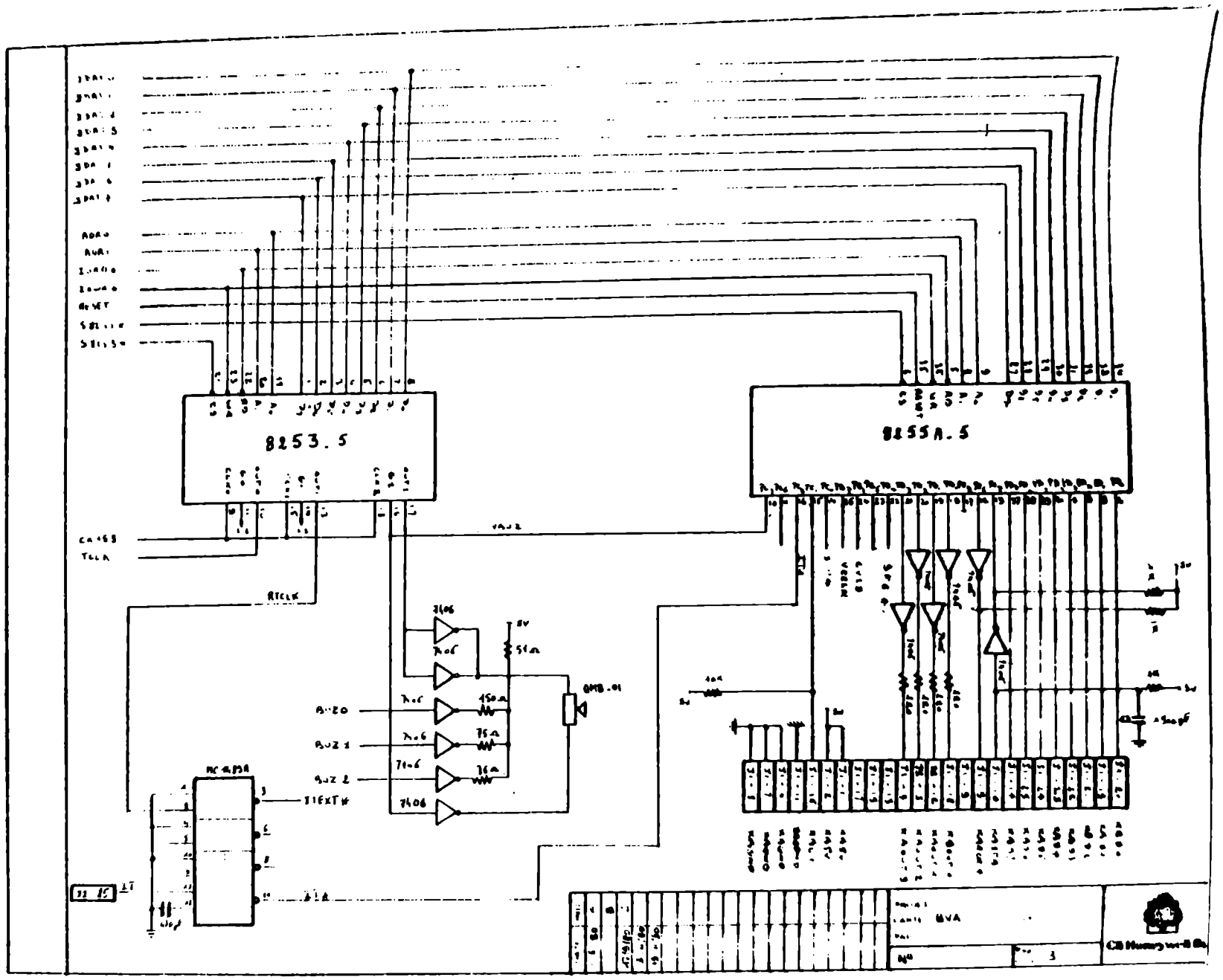
SCHEMAS ET CHRONOGRAMMES



MICROPROCESSEUR

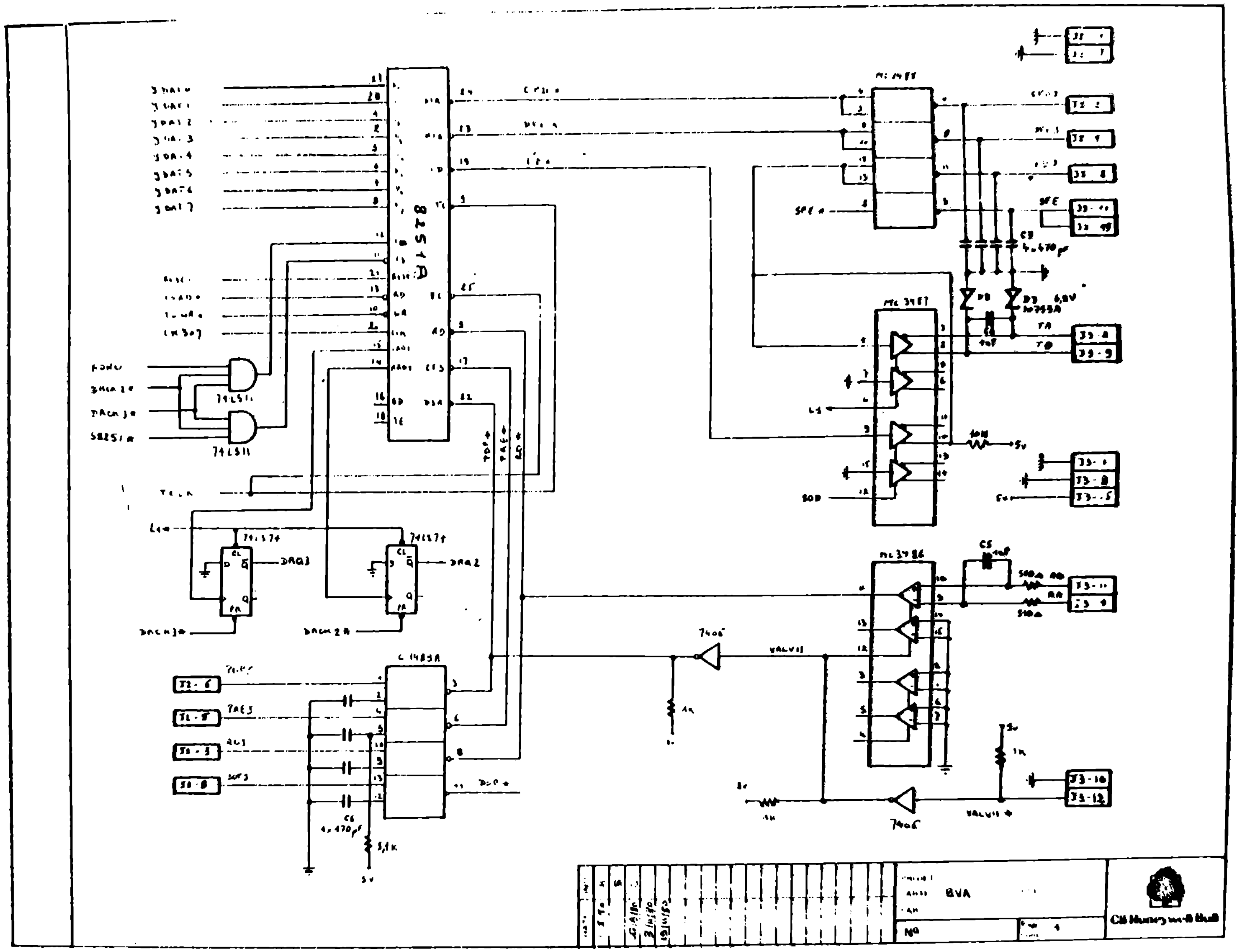


CONTROLEUR DMA

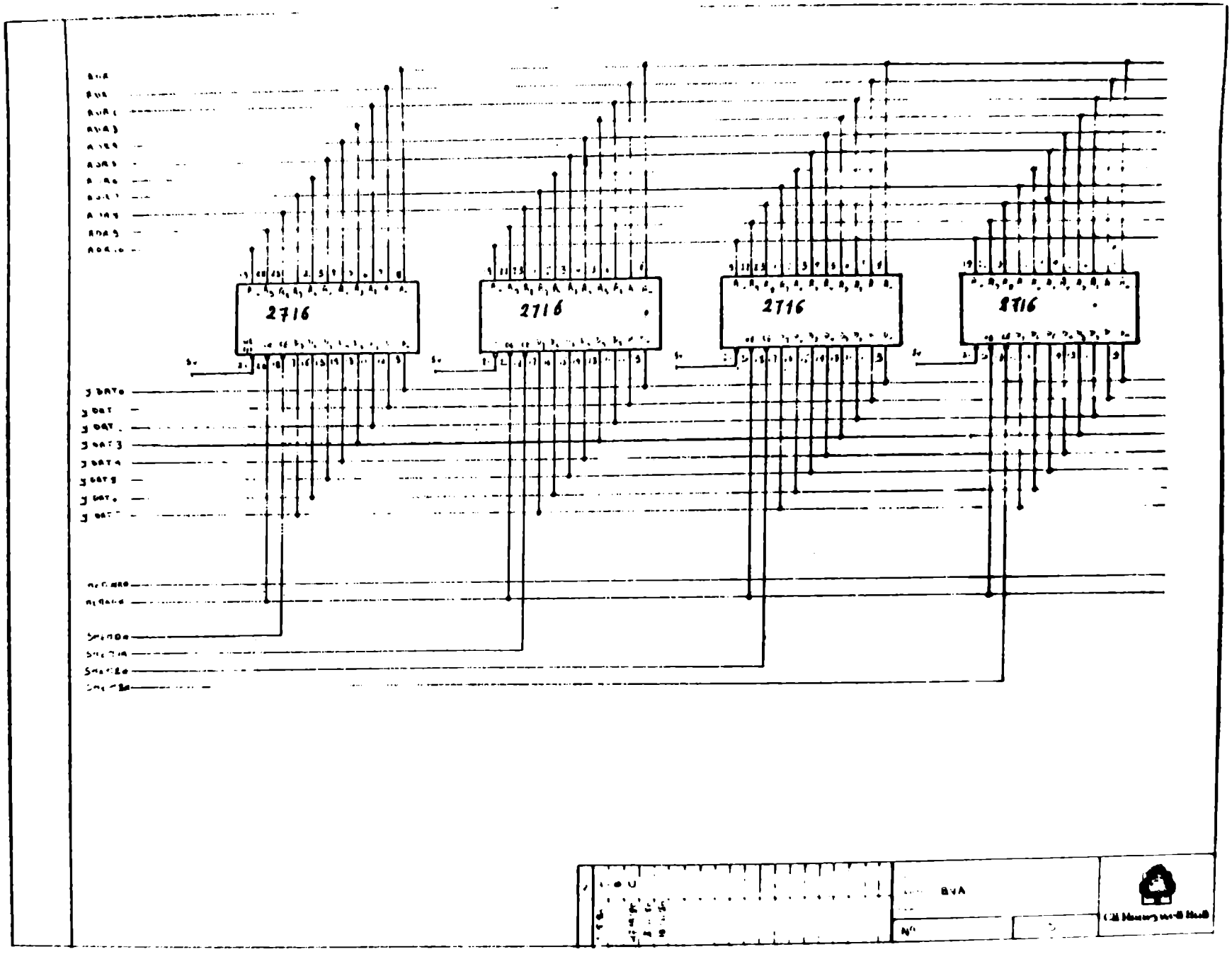


COMPTEURS

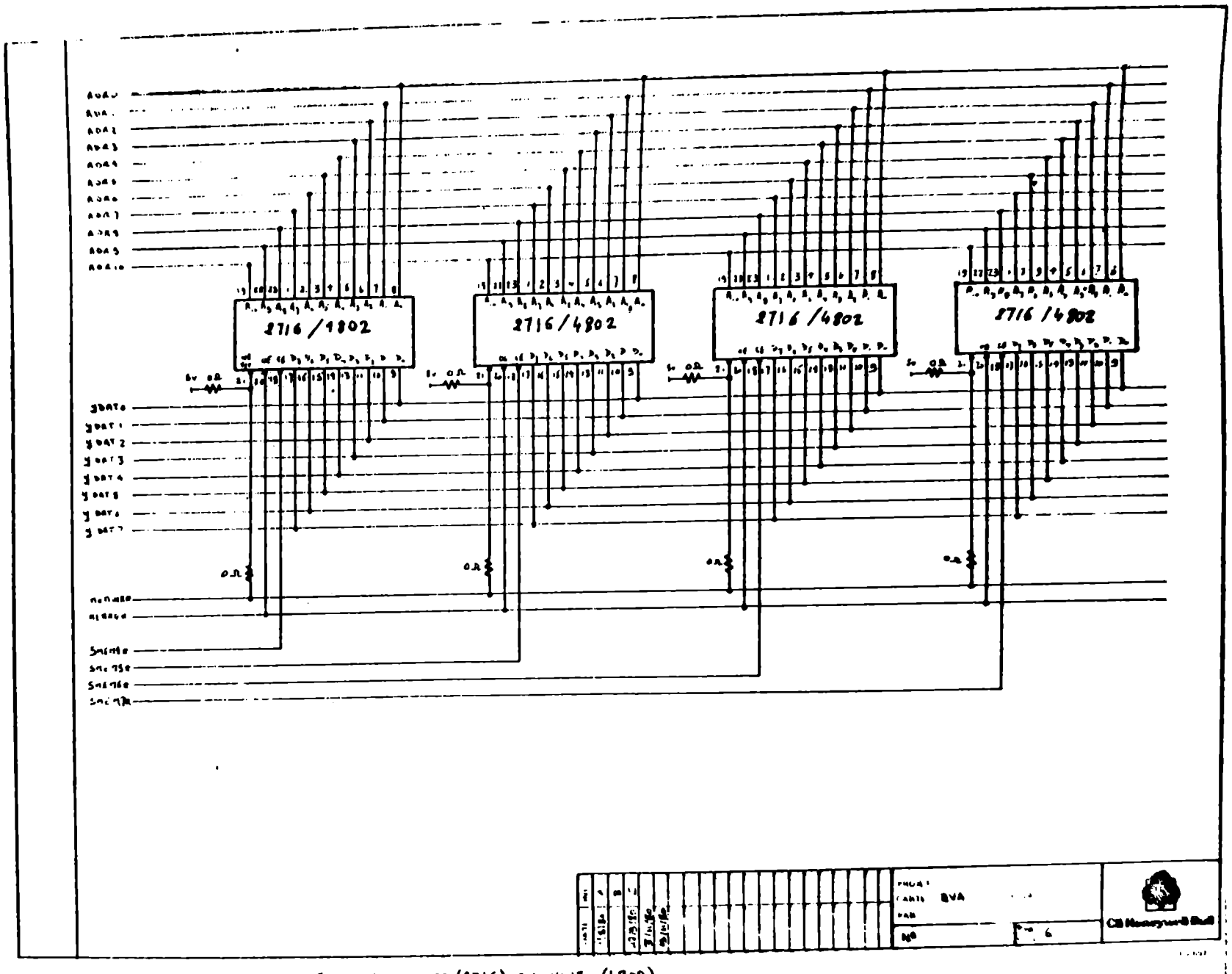
CONTROLEUR D'ENTREES/SORTIES (CLAVIER)



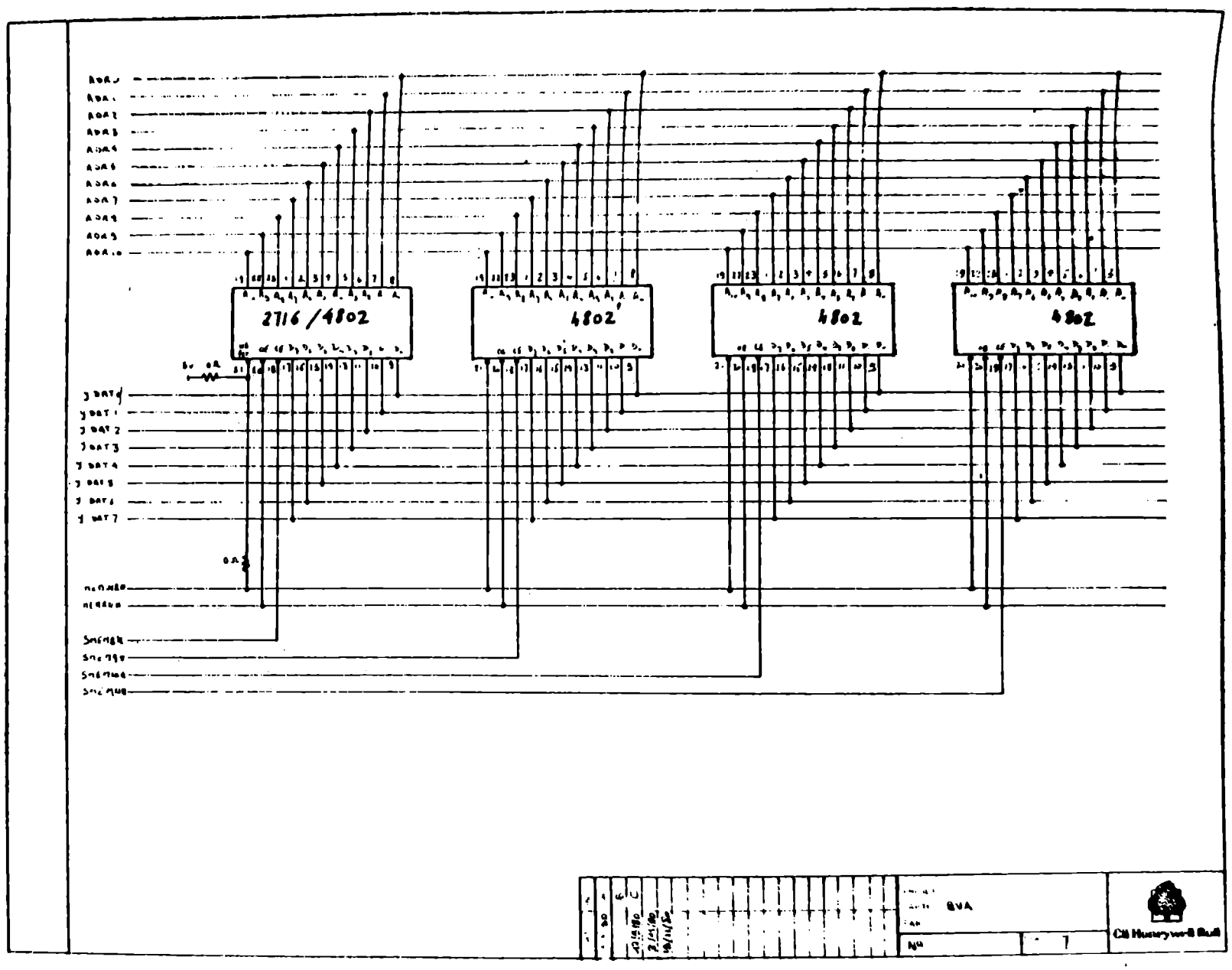
INTERFACE LIGNE DE COMMUNICATIONS



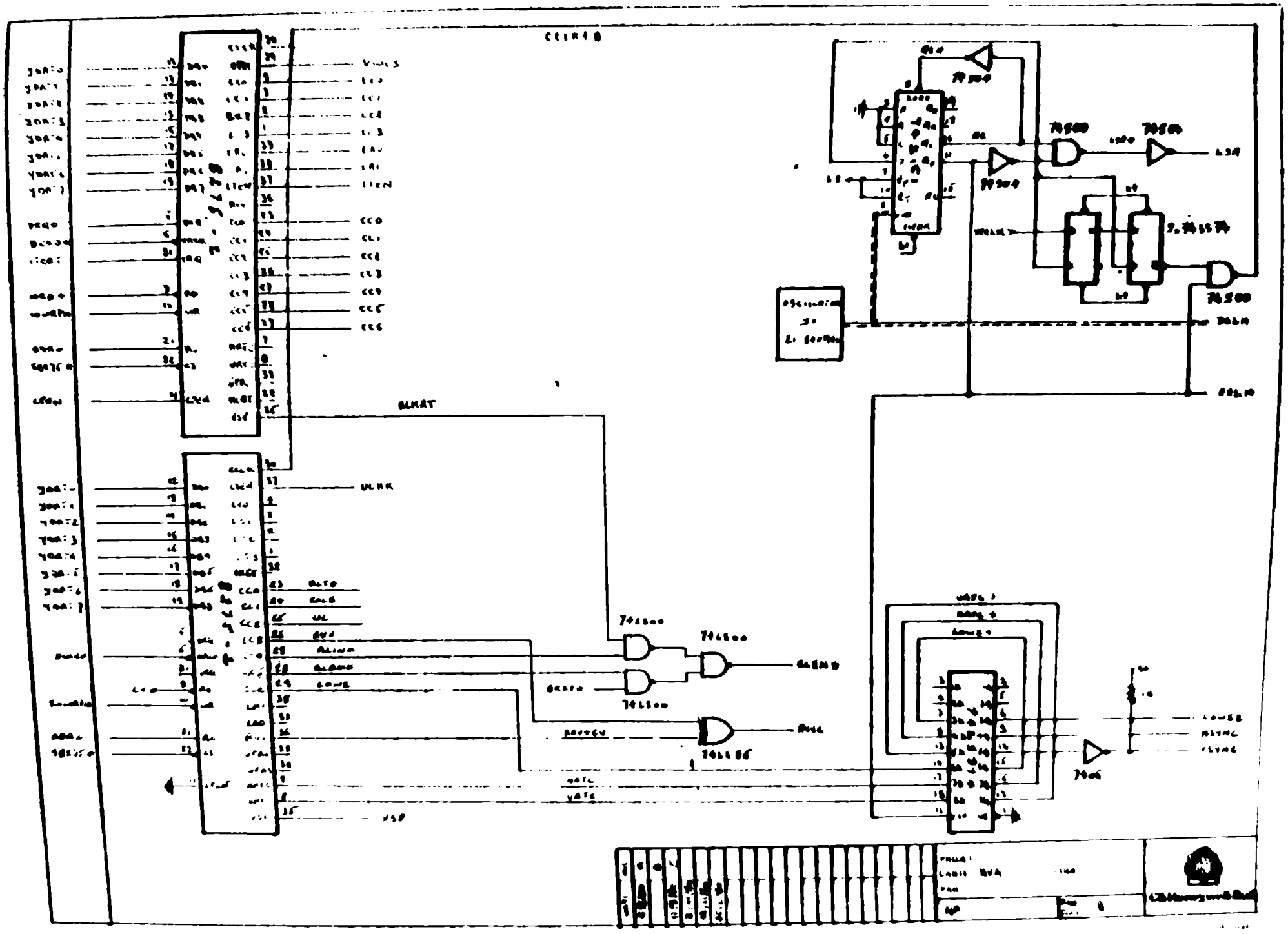
MEMOIRE DE COMMANDE



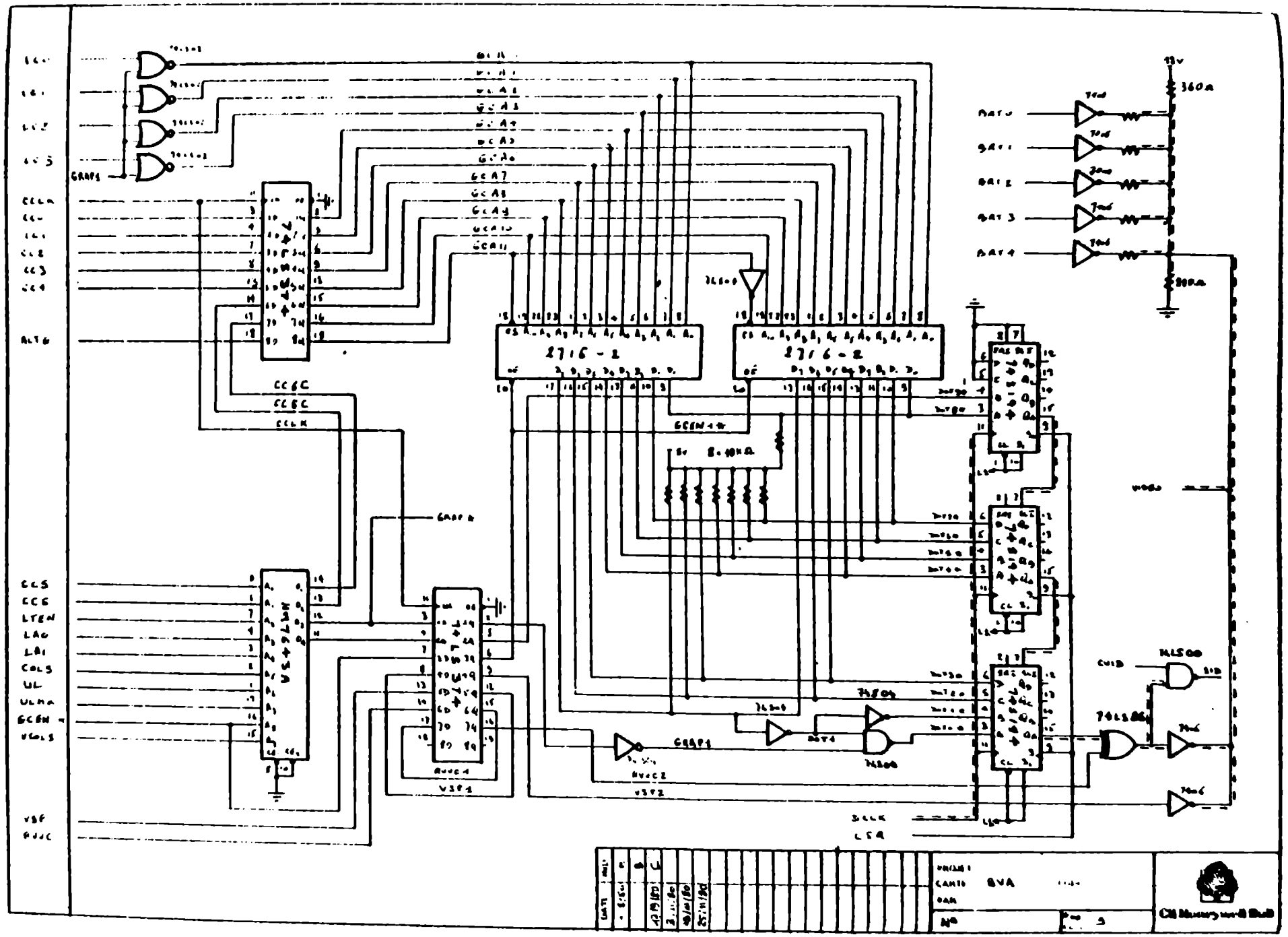
MÉMOIRE MORTE (2716) OU VIVE (4802)



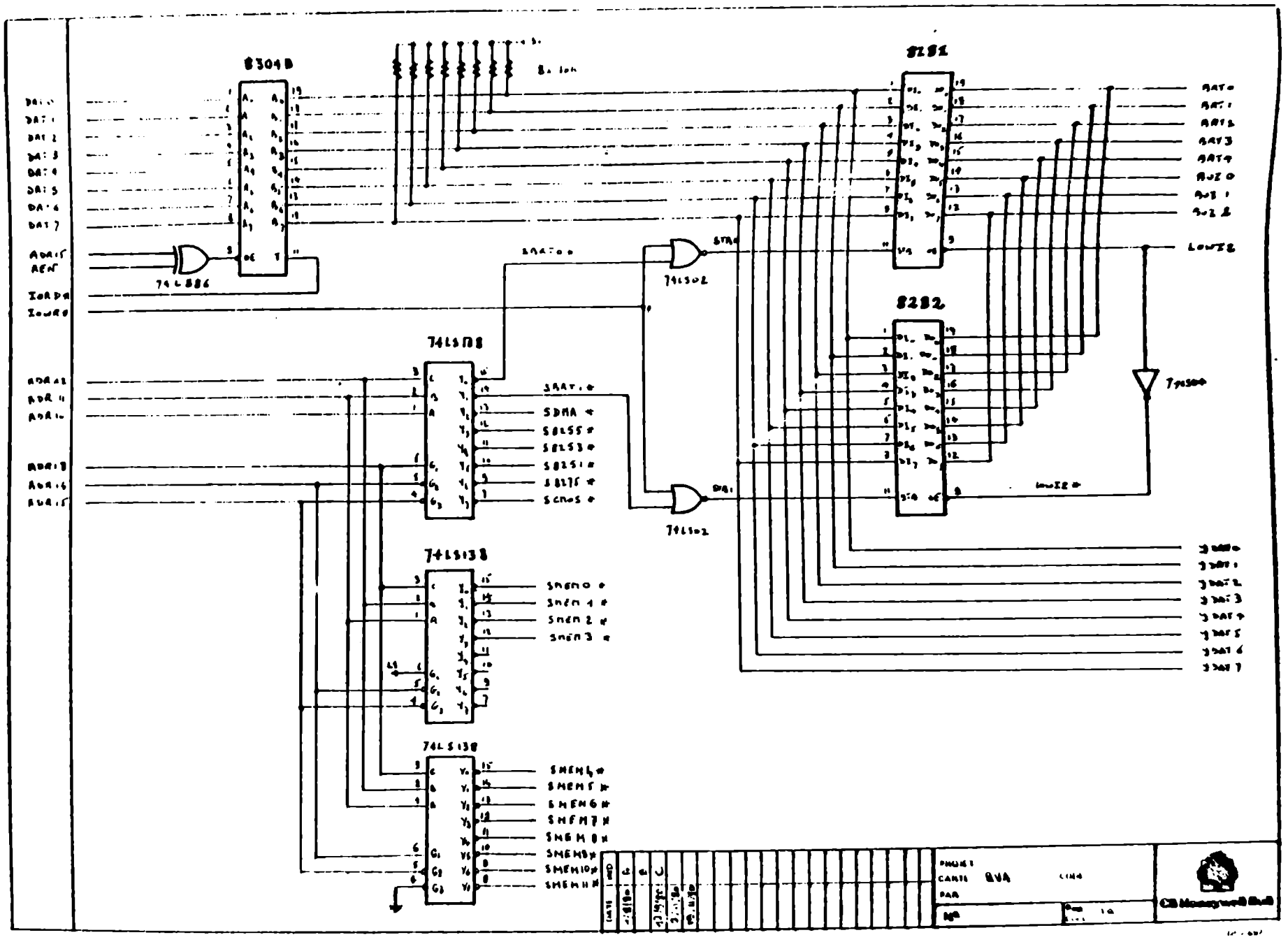
MEMOIRES DE TRAVAIL ET D'ECRAN



LOGIQUE DE VISUALISATION : CONTROLEURS DE CRT

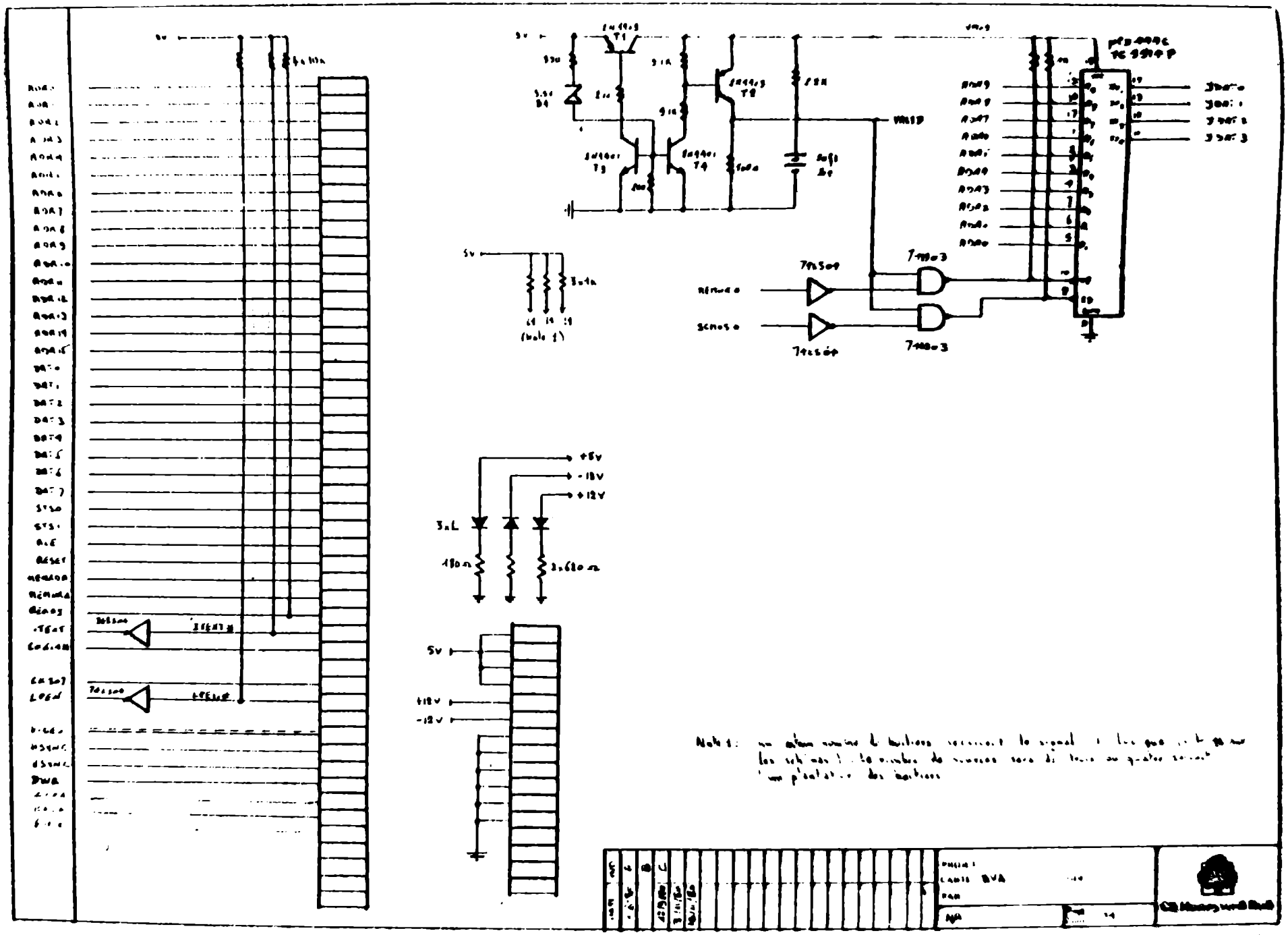


LOGIQUE DE VISUALISATION : GENERATEUR DE CARACTERES ET MIXAGES VIDEO



SELECTIONS DES CIRCUITS

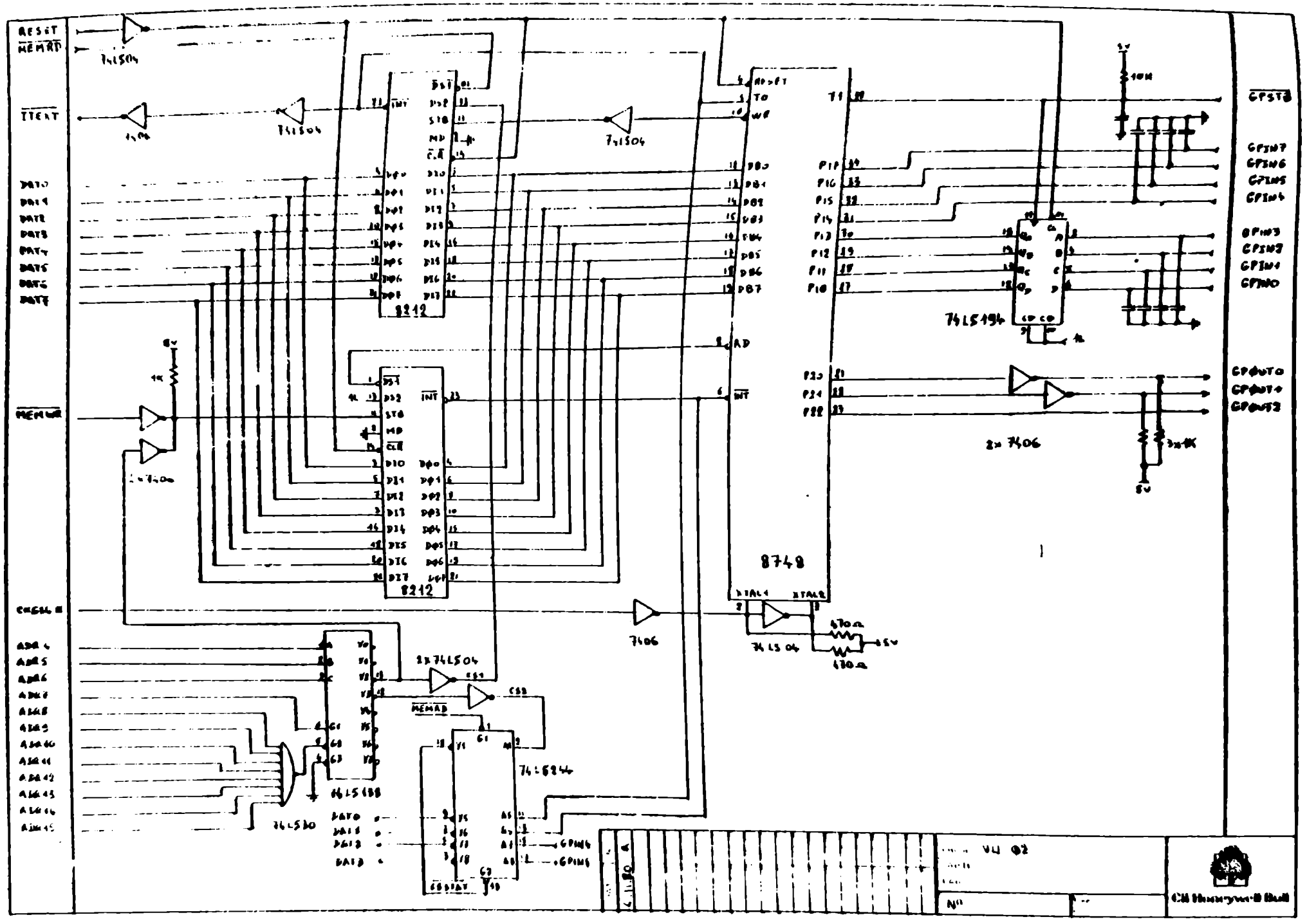
NIVEAUX LUMINEUX ET SONORES



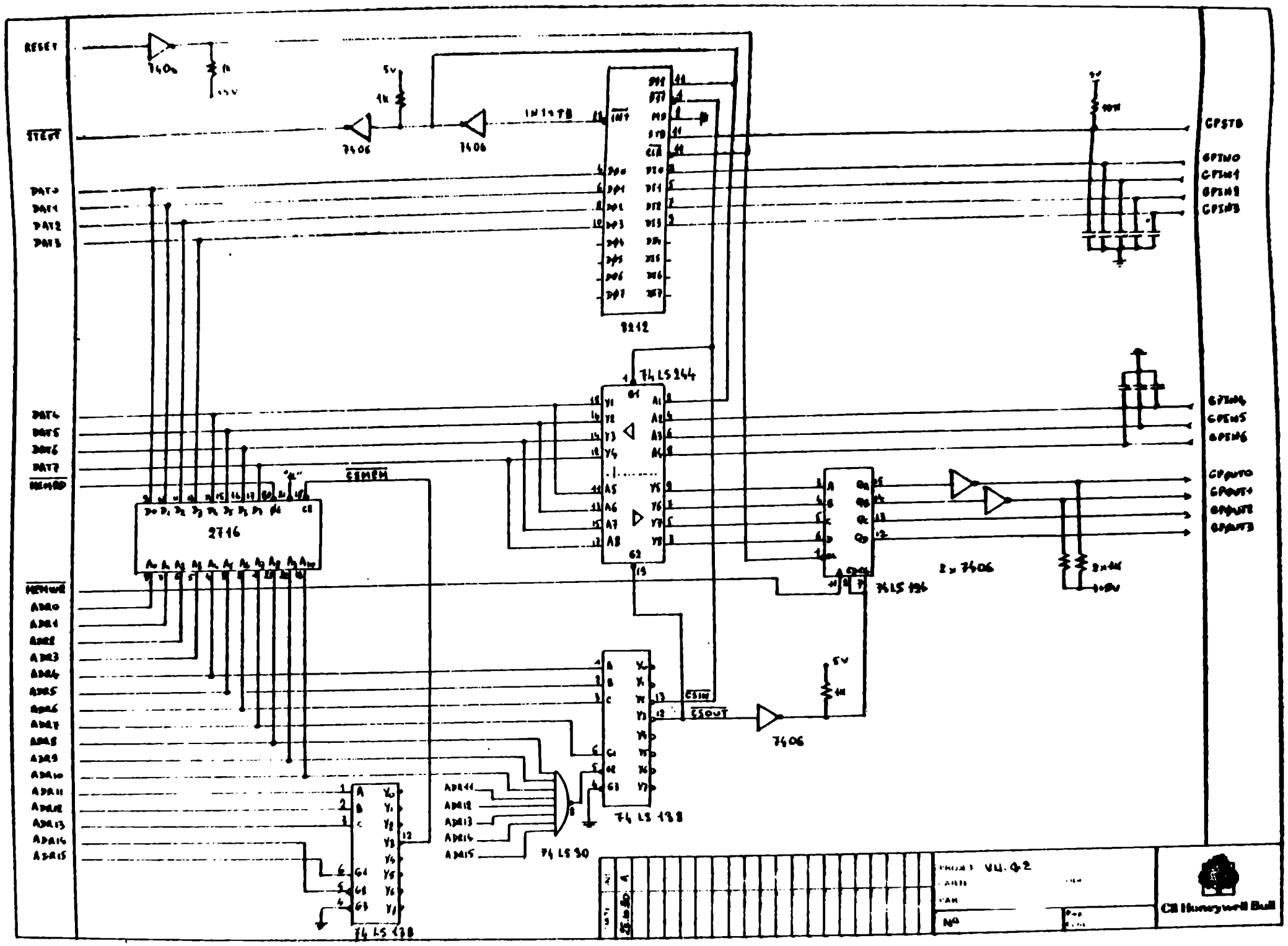
Note 1 : ...

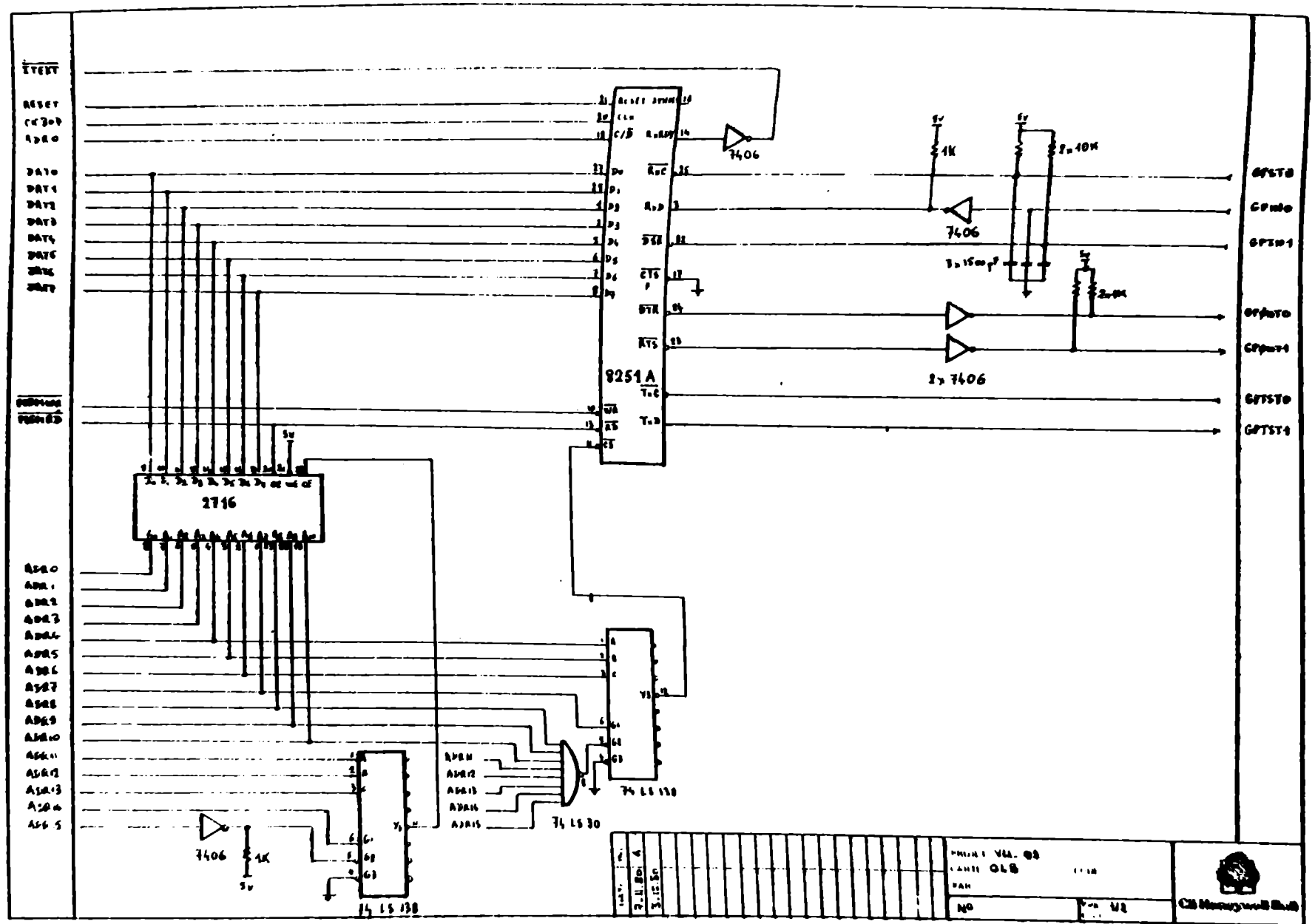
SIGNAUX D'INTERFACES OPTIONNELLES

SYSTEME DE SAUVEGARDE

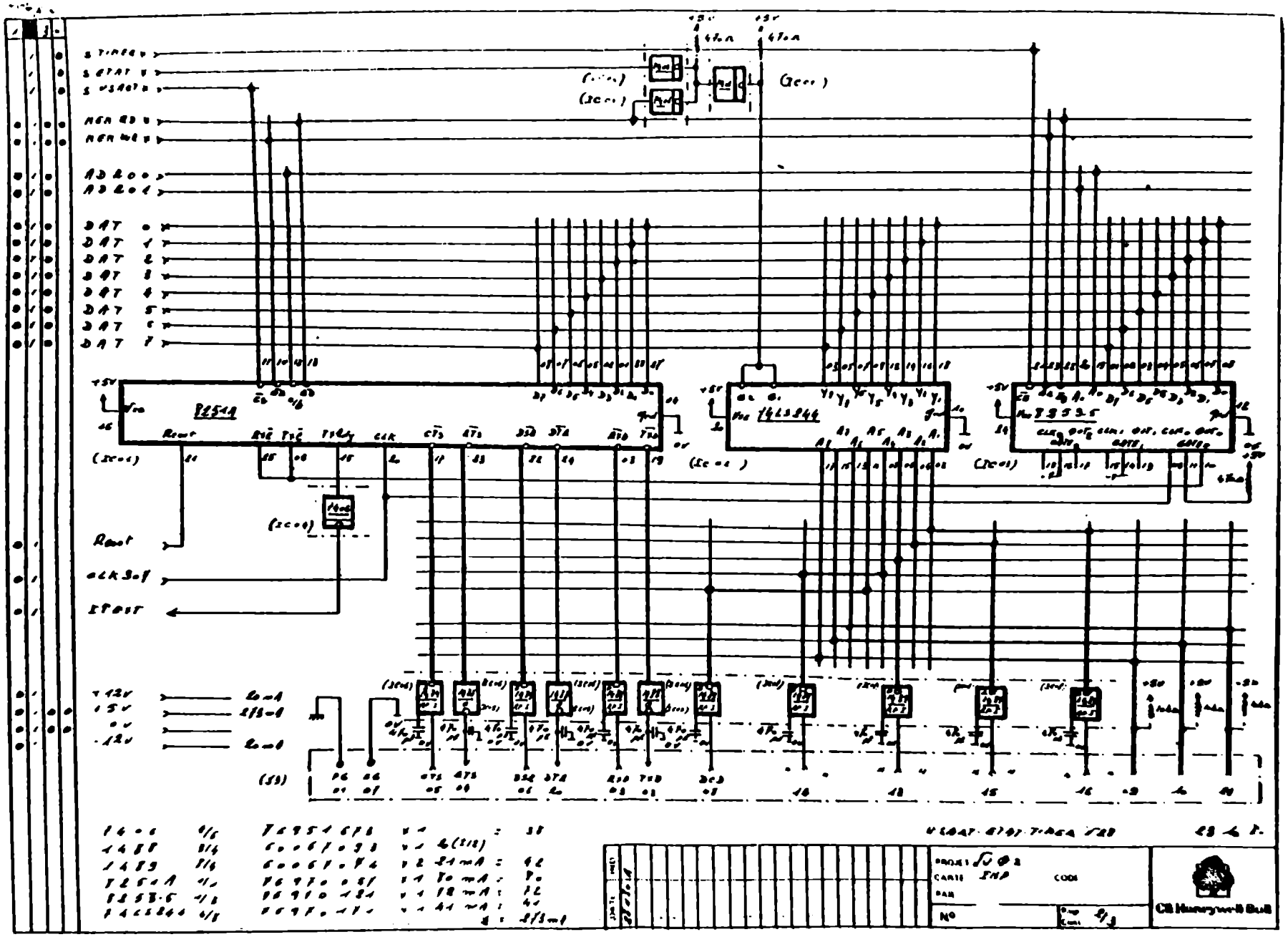


INTERFACE LECTEURS DE BADGES, DOCUMENTS,... (PRINCIPE 1)

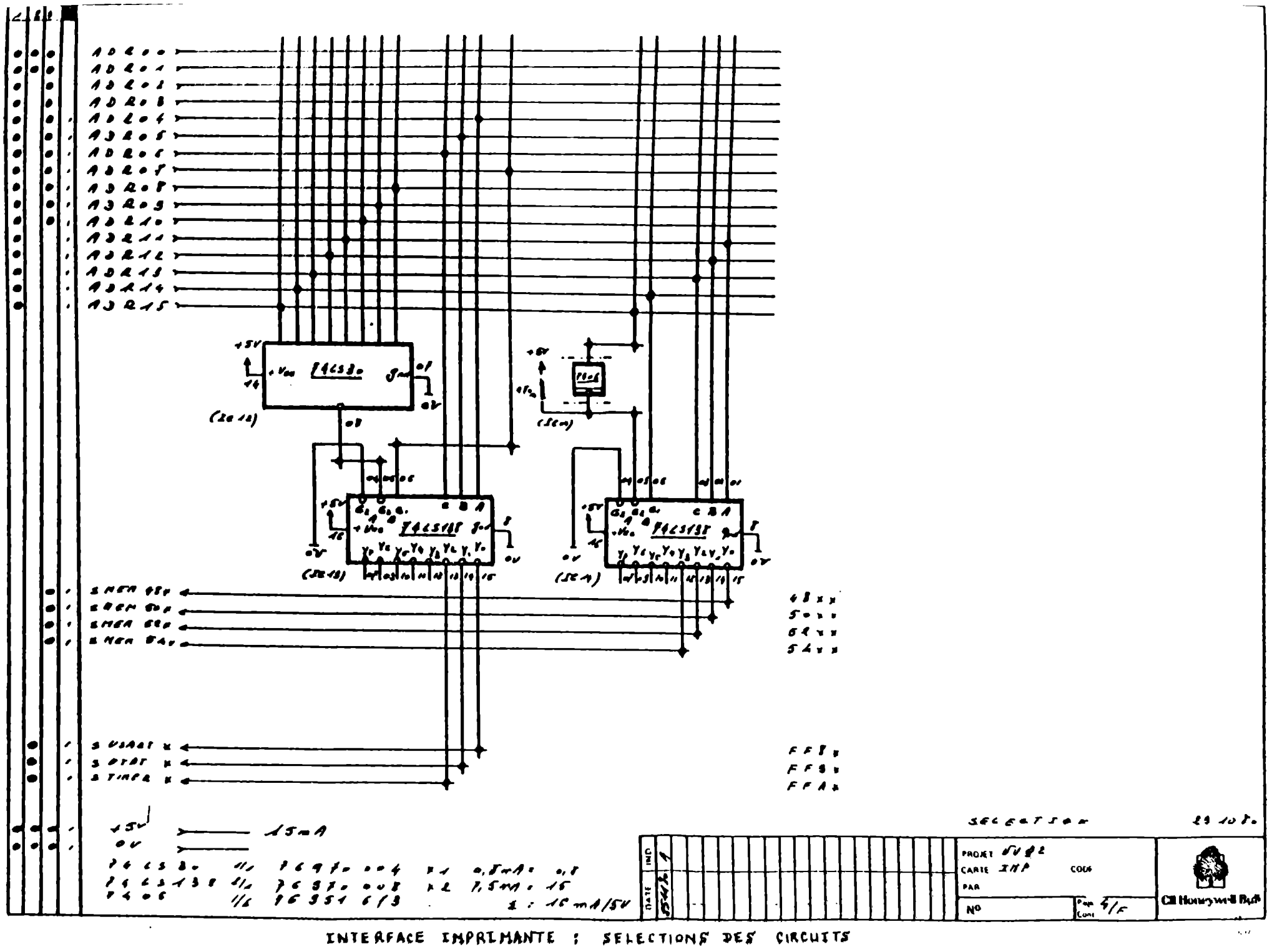

RESET
STEST
DAT0
DAT1
DAT2
DAT3
DAT4
DAT5
DAT6
DAT7
MEMRD
MEMWR
ADR0
ADR1
ADR2
ADR3
ADR4
ADR5
ADR6
ADR7
ADR8
ADR9
ADR10
ADR11
ADR12
ADR13
ADR14
ADR15
7406
5V
1k
8212
74LS244
2716
74LS138
74LS30
74LS196
2 x 7406
GPSTB
GPIN0
GPIN1
GPIN2
GPIN3
GPIN4
GPIN5
GPIN6
GPOUT0
GPOUT1
GPOUT2
GPOUT3
CSMEM
CSIN
CSOUT
PROJECT VU.42
No
CII Honeywell Bull



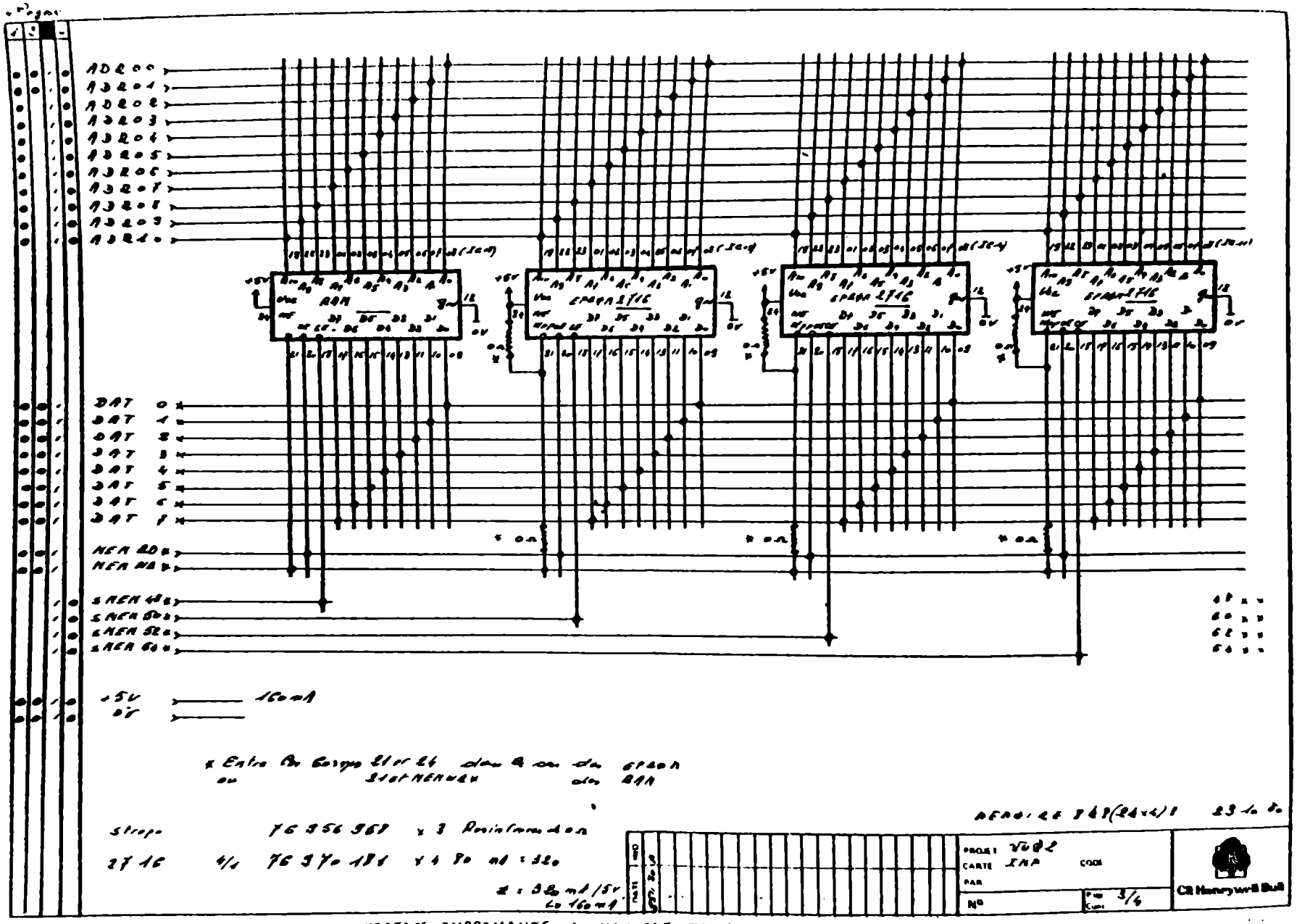
INTERFACE LECTEUR DE BADGES (PRINCIPE 2)



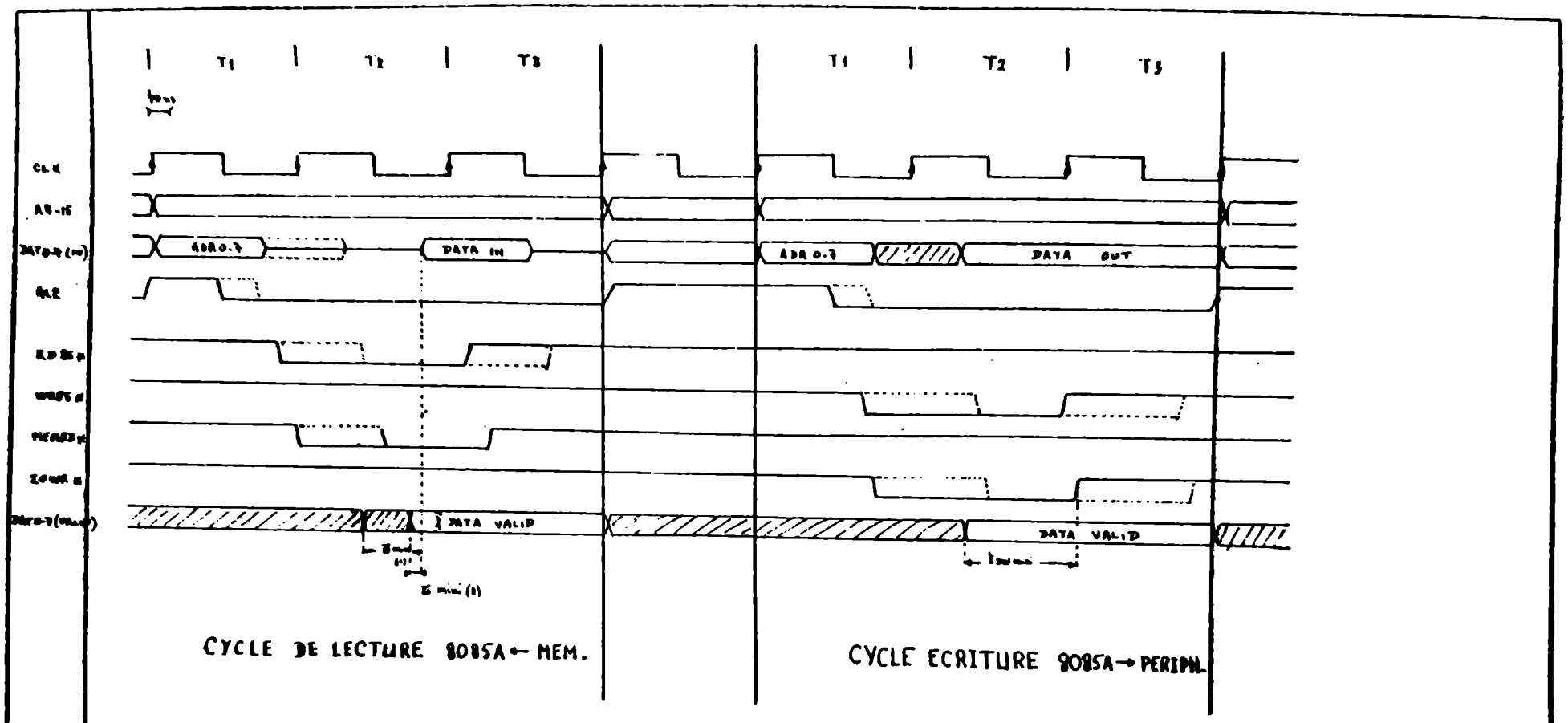
INTERFACE IMPRIMANTE : INTERFACE LIGNE ET GENERATEUR DE VITESSES



INTERFACE IMPRIMANTE : SELECTIONS DES CIRCUITS



INTERFACE IMPRIMANTE : MEMOIRE TAMPON ET MEMOIRE DE COMMANDE

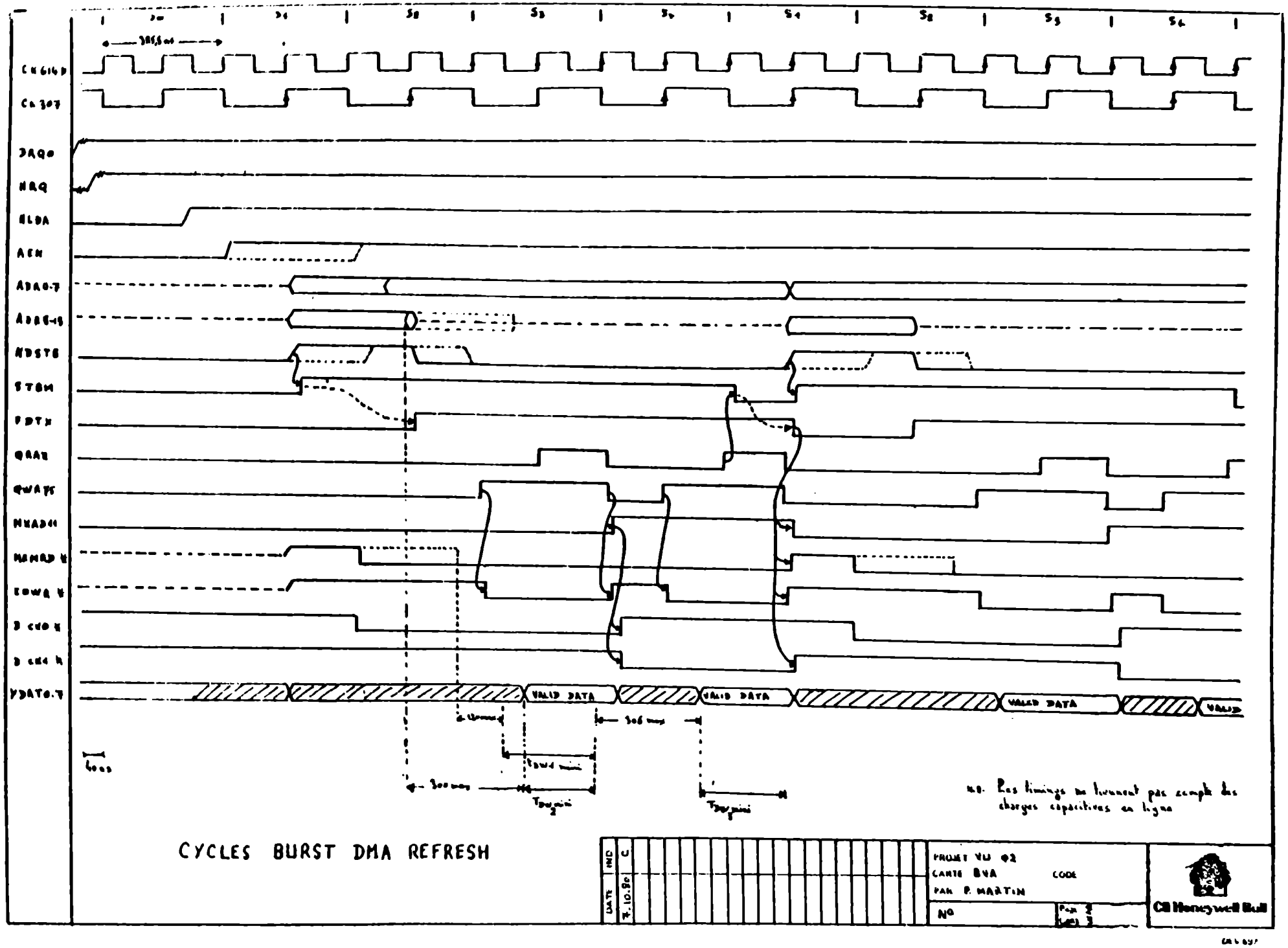


(1) Timing (avec T_{cyc} = 8.15 ns sur le 8085A) de la chaîne "MEMRD"
(2) Timing de la chaîne "SMEMRD"

N.B. Les timings ne prennent pas en compte les charges capacitives en ligne

PROJET VU.02		
CARTE 8VA		
PAR P. MARTIN		
N°	Page 1	
	Cadre 3	

CII Honeywell Bull

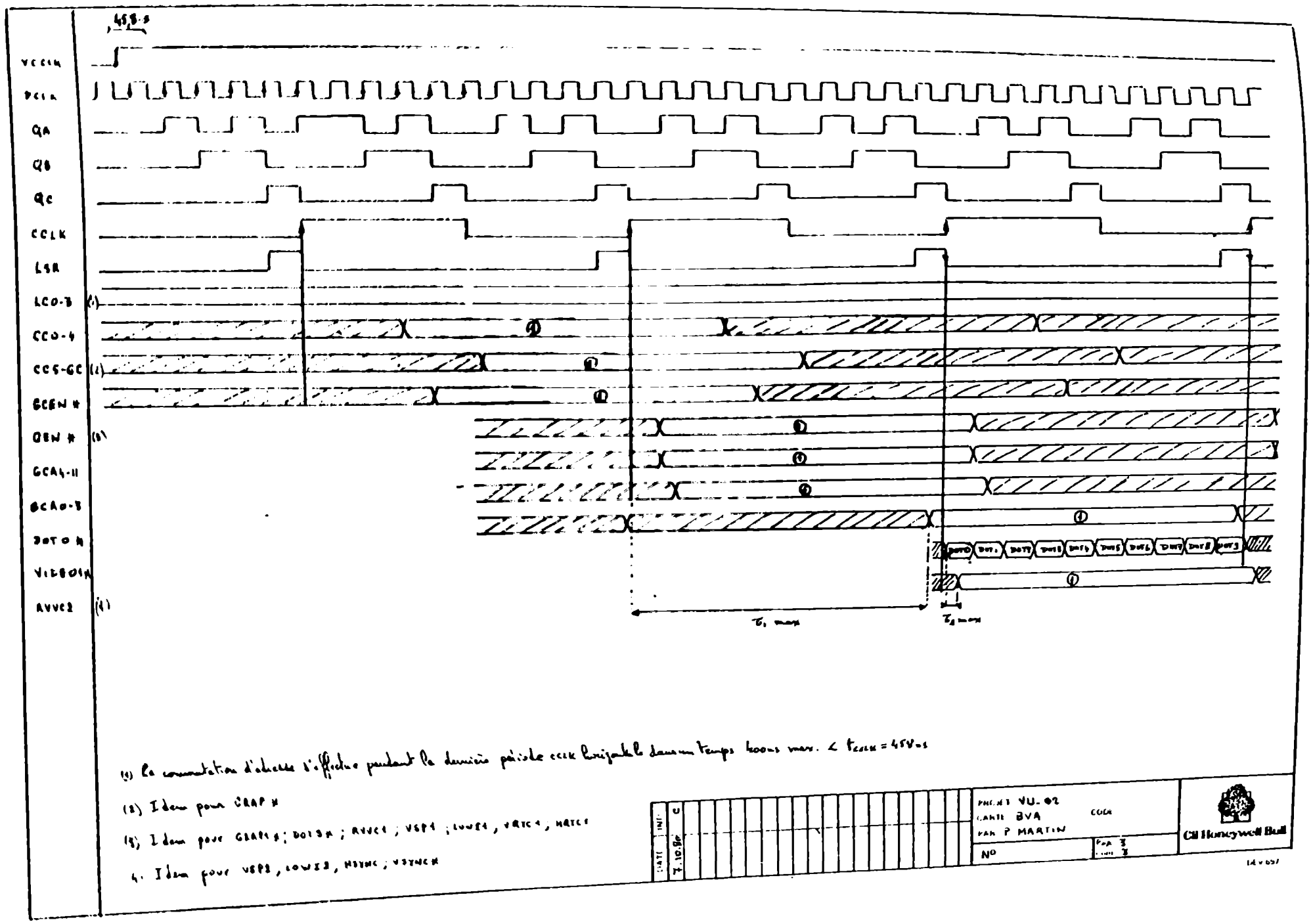


CYCLES BURST DMA REFRESH

DATE	IND	C
7.10.80		

PROJET VU Φ2
CARTE BVA CODE
PAR P. MARTIN
N° Page 2 Cadre 3

CII Honeywell Bull



(1) La commutation d'adresse s'effectue pendant la dernière période CCLK horizontale dans un temps 400ns max. < t_{CCLK} = 458ns

(2) Idem pour GRAP *

(3) Idem pour GRAP1 *; DOT3 *; RVVC1; VSP1; LOWI1, VRTC1, HRTC1

4. Idem pour VSP2, LOWI2, HSYNC; VSYNC *

DATE	INI	
7.10.80	C	

PROJET VU-02
CARTE BVA
PAR P MARTIN
CODE
N°
PAR 3
COM 3

Cii Honeywell Bull

14 v 657